FIG. 1

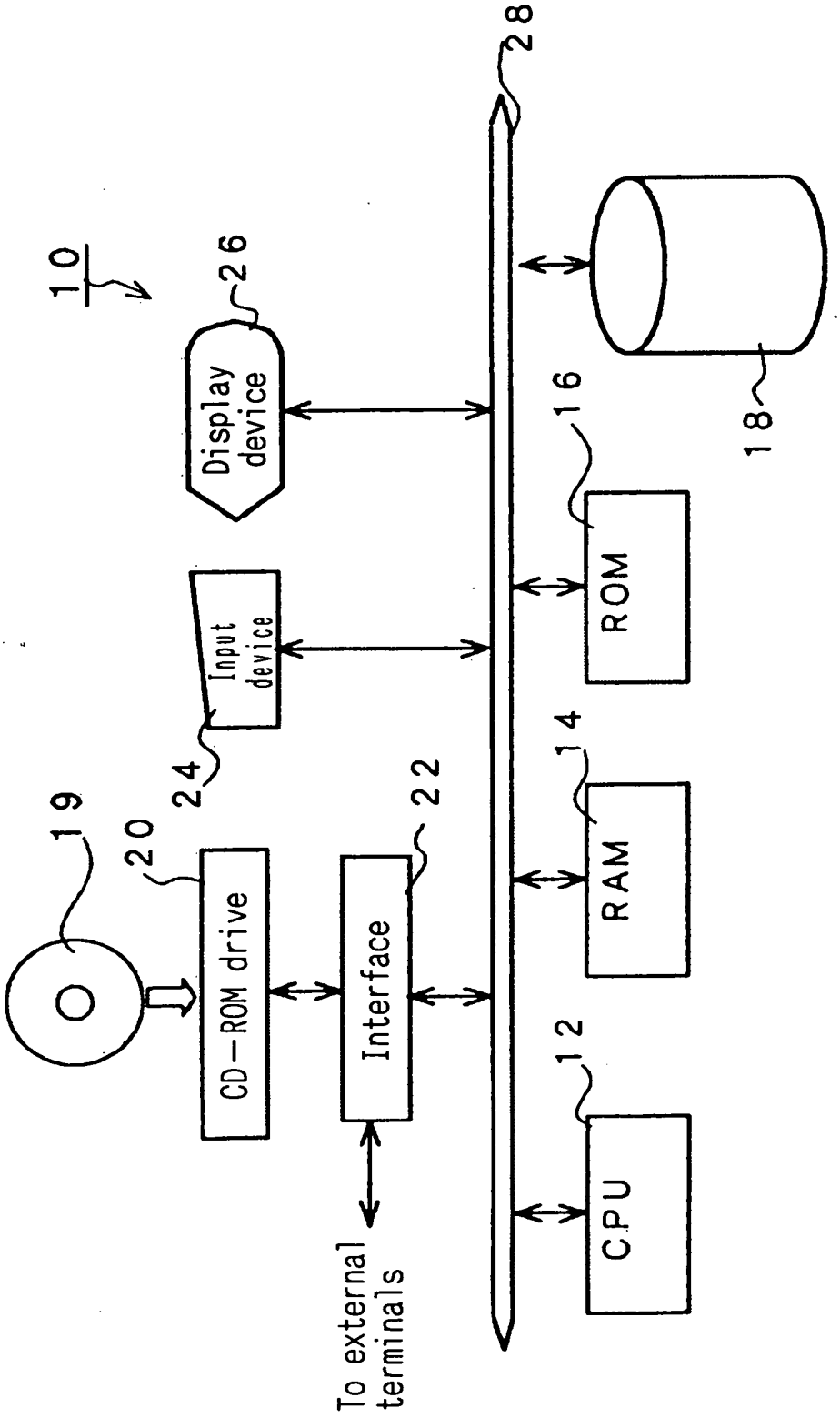

10
28
26
Display device
16
ROM
18
24
Input device
19
20
22
14
RAM
CD-ROM drive
Interface
12
CPU
To external terminals

FIG. 2

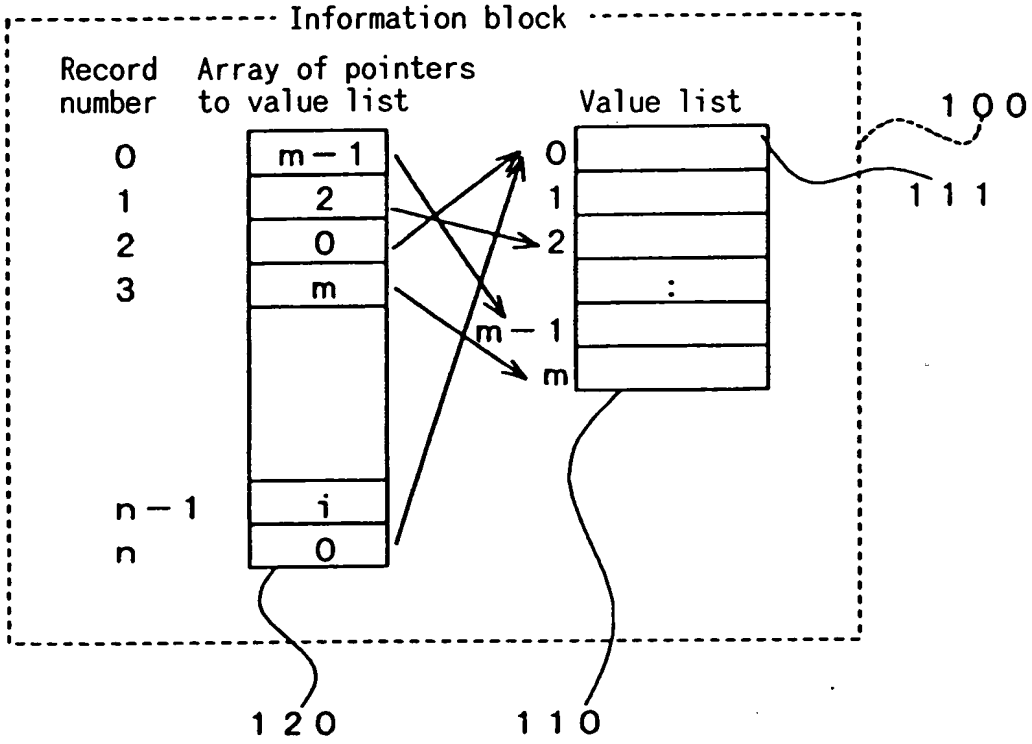

Information block
Record number
Array of pointers to value list
Value list
100
111
0
1
2
3
n−1
n
m−1
2
0
m
i
0
0
1
2
:
m−1
m
120
110

FIG. 3A

Record No.	Customer ID	Customer name	Telephone number
0	1	Smith	333-aaa
1	2	Lennon	441-bbbb
2	3	Bernstein	558-cccc
3	4	Carlson	998-dddd
4	5	Ford	112-eeee
5	6	Eagle	225-ffff

FIG. 3B

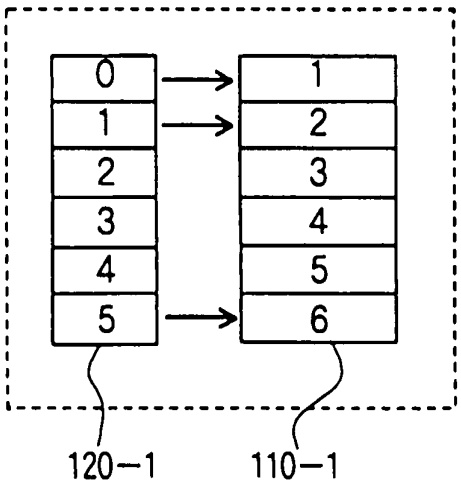


FIG. 3C

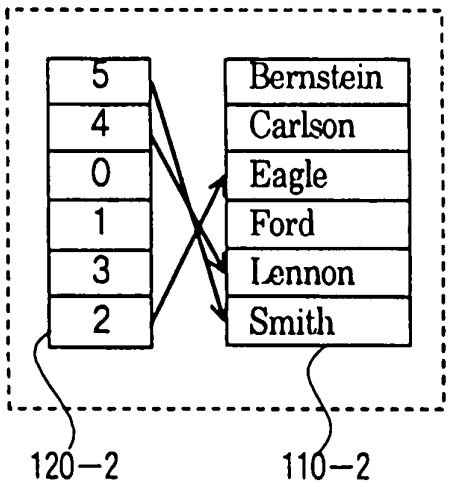


FIG. 3D

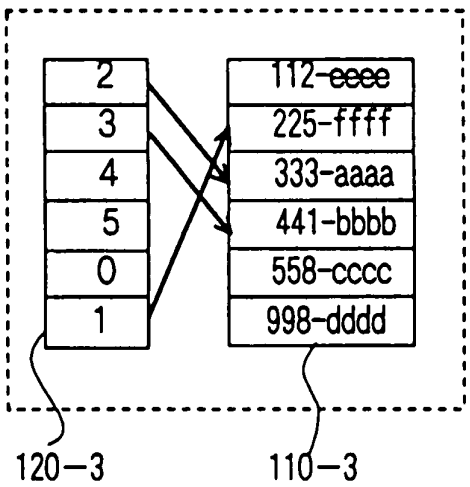


FIG. 4A

Record No.	Sex	Age	Occupation
0	Female	18	Programmer
1	Male	21	Student
2	Female	31	Teacher
:	:	:	:
:	:	:	:
999999	Female	16	Student

FIG. 4B

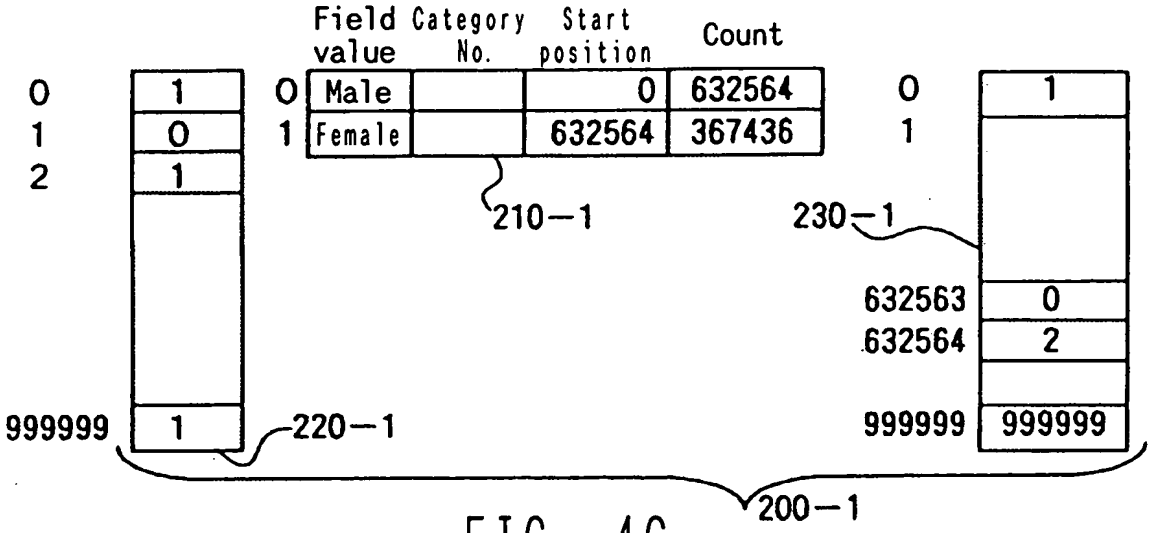


FIG. 4C

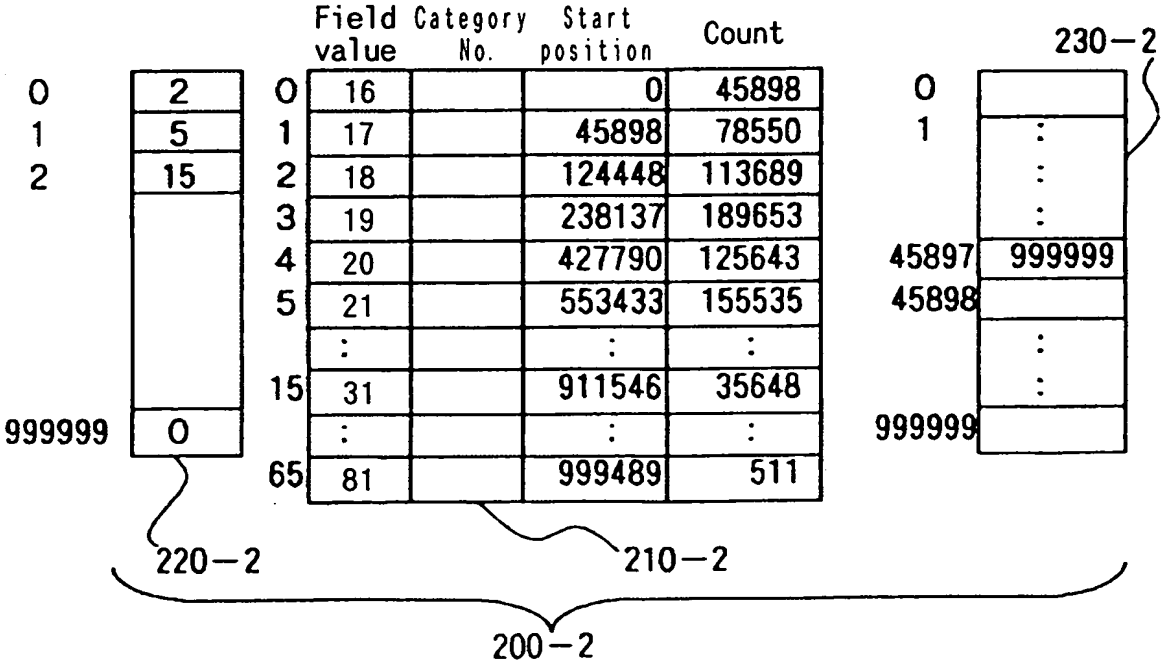
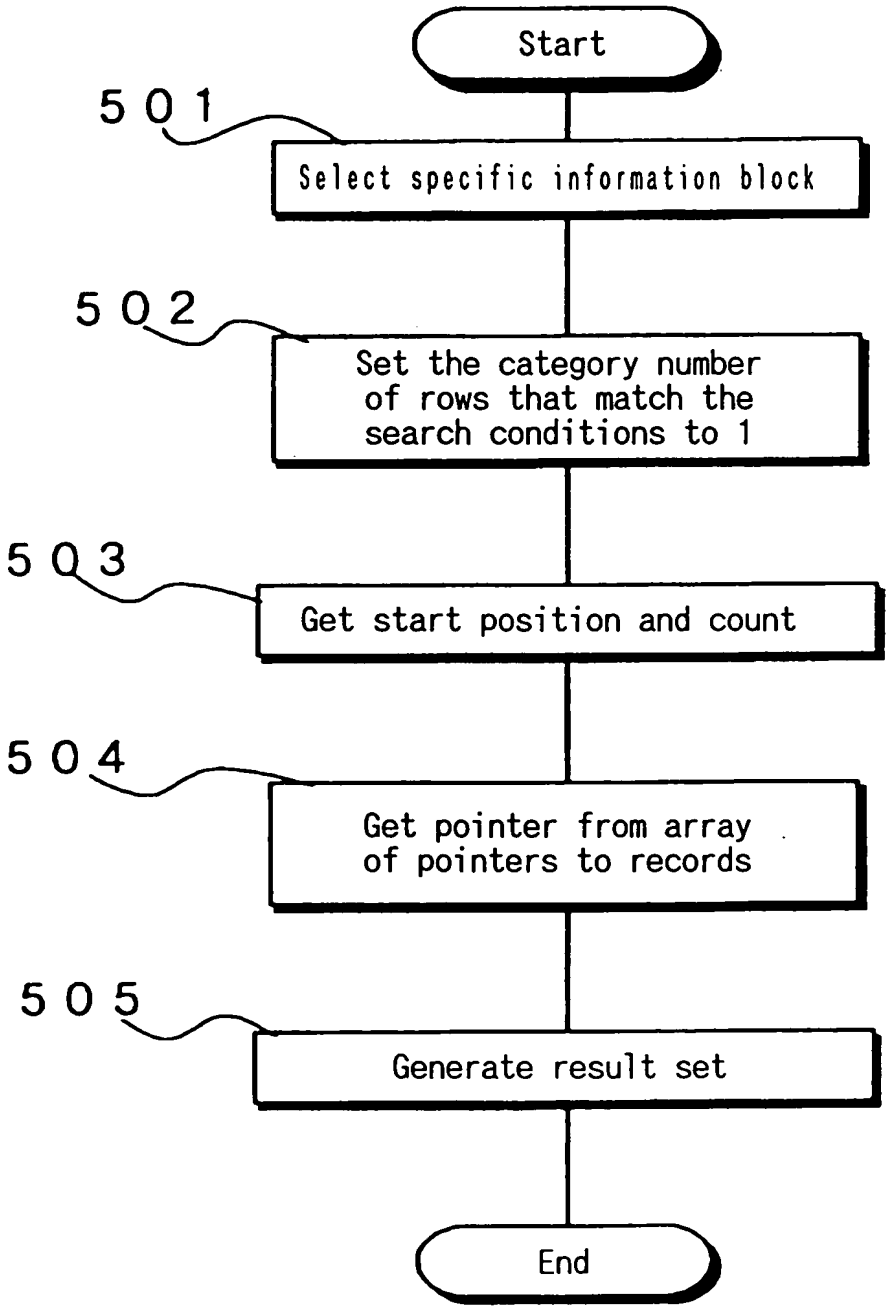
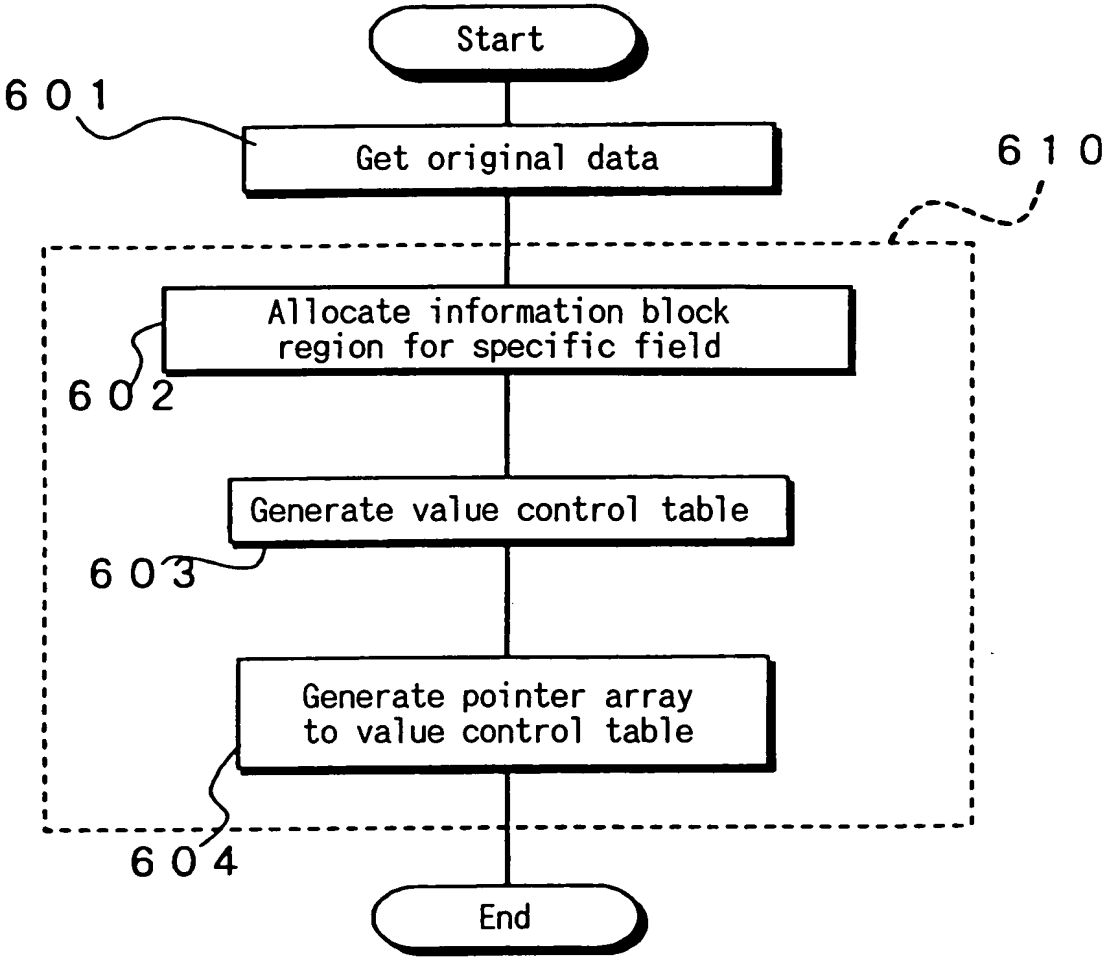


FIG. 5



HOSTED WITH OTTO

FIG. 6



EFS ID: 1748

FIG. 7A

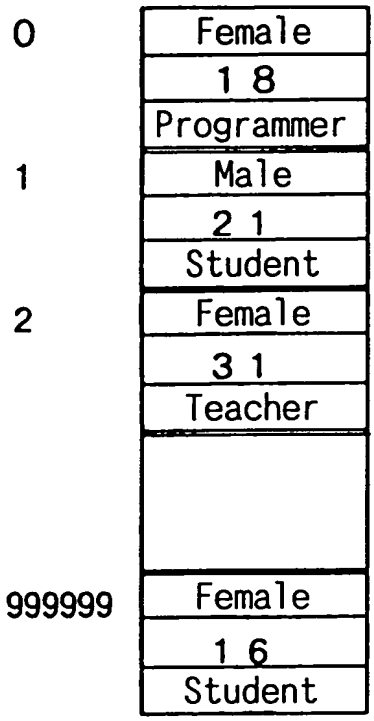
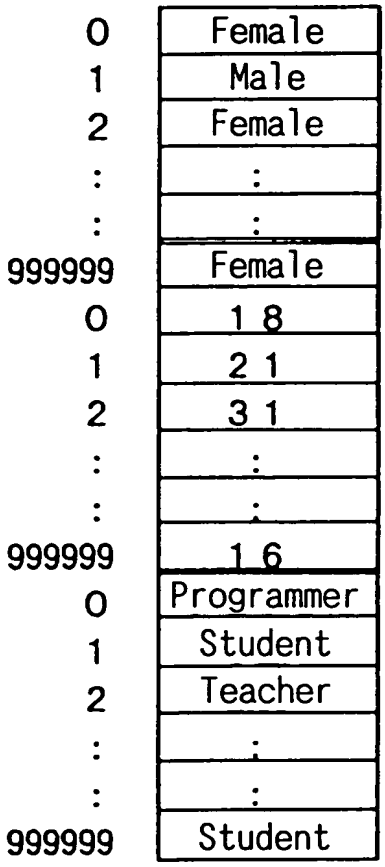


FIG. 7B



JP020048231-070204

09860.P.62384.DOT

FIG. 8A

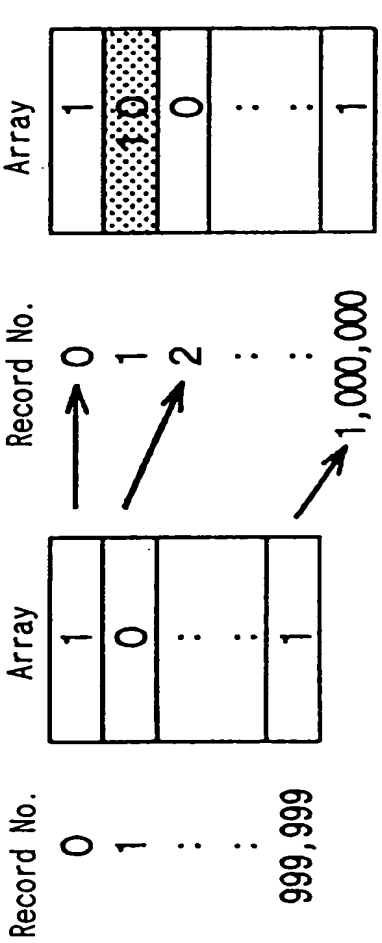


FIG. 8B

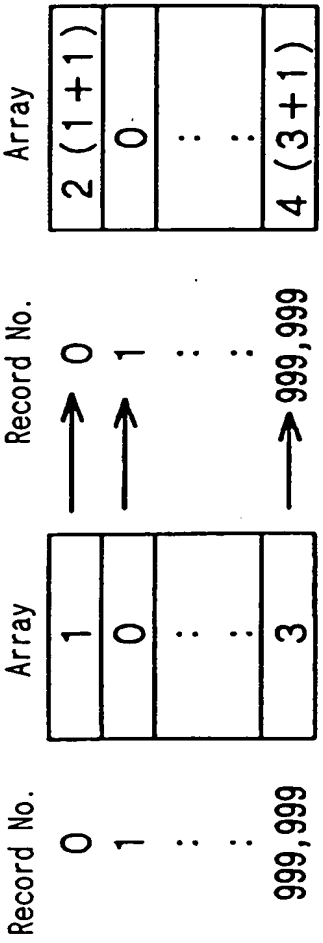
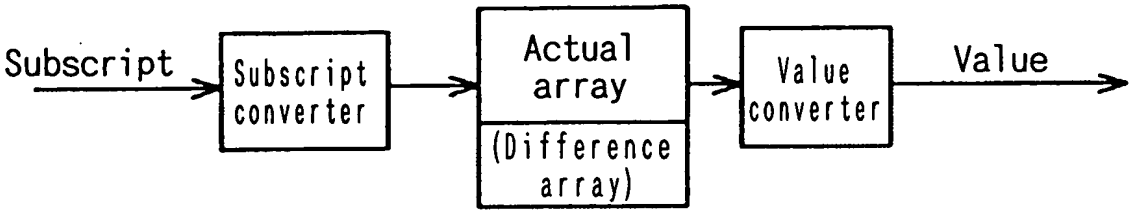


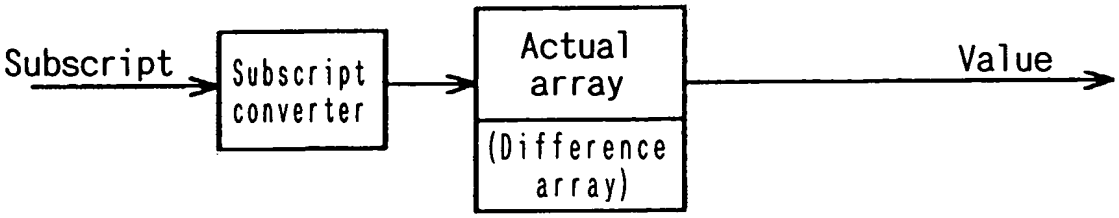
FIG. 9A

FIG. 9B



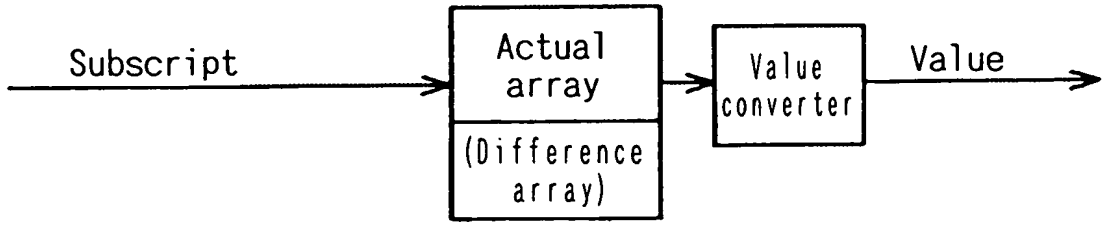
Example of converting both subscripts and values

FIG. 9C



Example of converting subscripts only

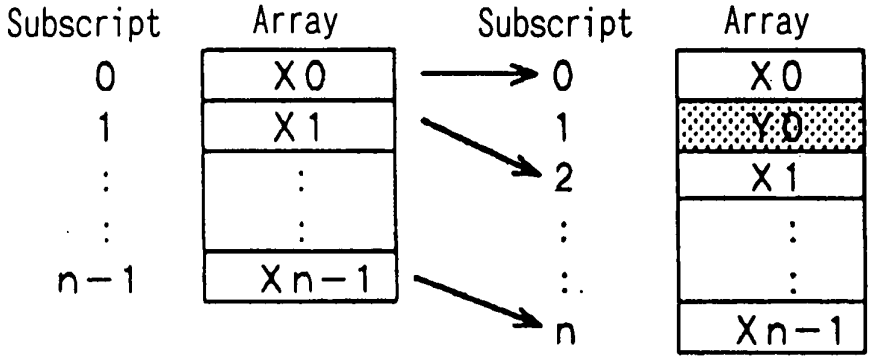
FIG. 9D



Example of converting values only

EFS Web 2.2.4

FIG. 10A



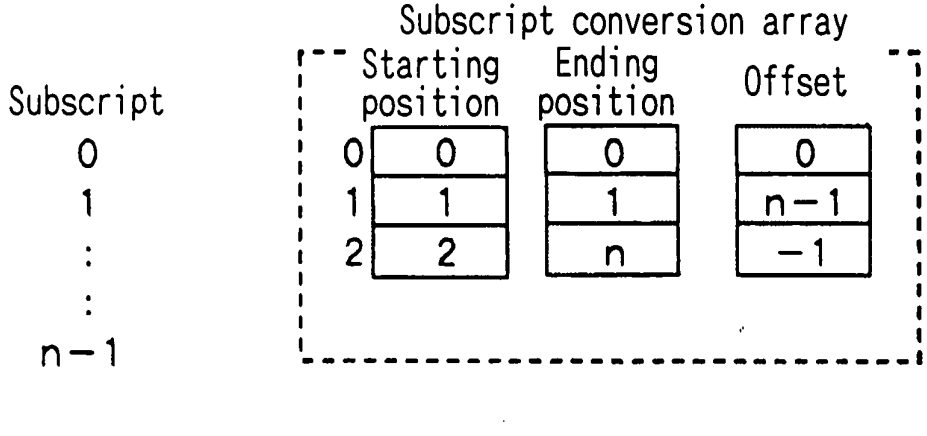
Insertion of an element into an array of size "n"

FIG. 10B

Subscript	Stored value
0	X0
1	Y0
2	X1
:	:
:	:
n	Xn−1

Logical relationship between subscripts and stored values

FIG. 10C



10/048231

FIG. 11

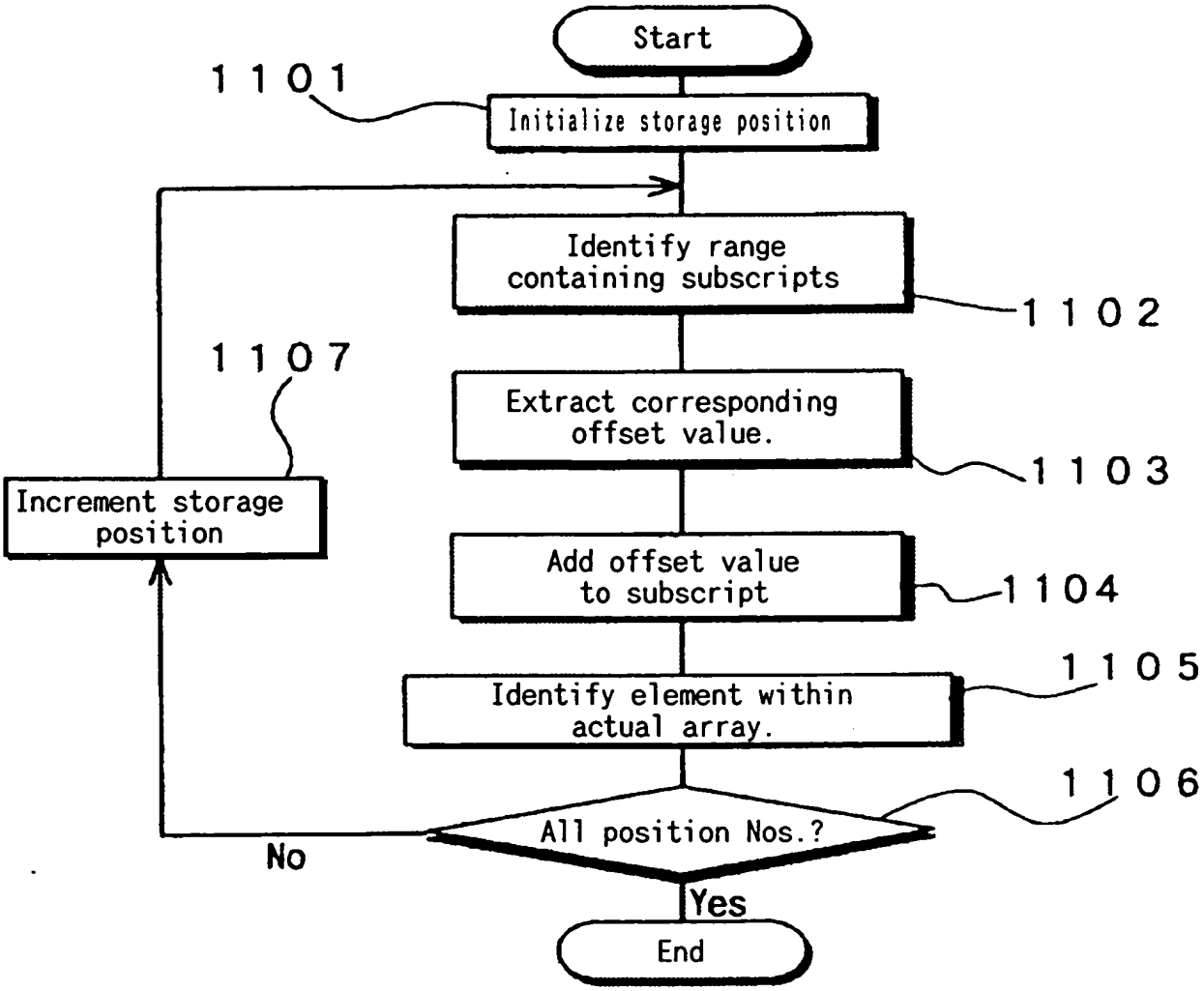

Start
1101
Initialize storage position
Identify range containing subscripts
1102
1107
Extract corresponding offset value.
1103
Increment storage position
Add offset value to subscript
1104
1105
Identify element within actual array.
1106
All position Nos.?
No
Yes
End

FIG. 12

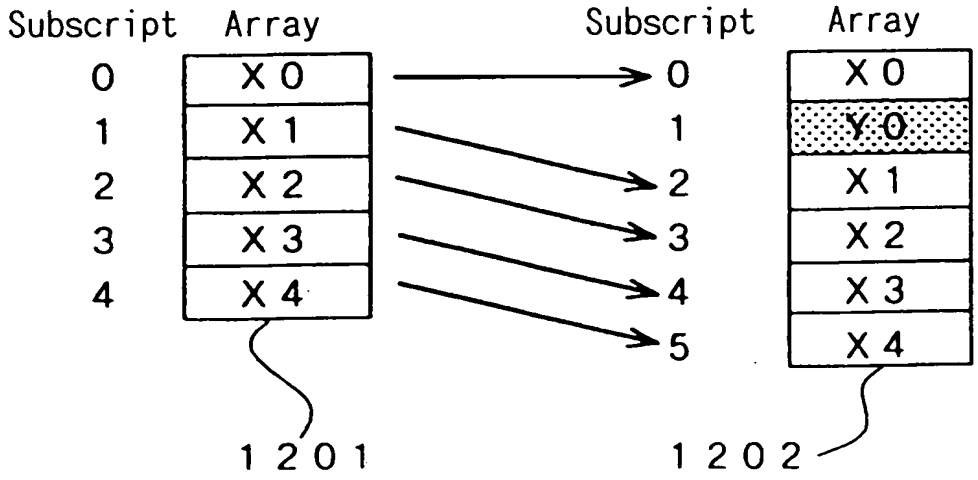

Subscript
Array
0
1
2
3
4
X0
X1
X2
X3
X4
1201
Subscript
Array
0
1
2
3
4
5
X0
Y0
X1
X2
X3
X4
1202

FIG. 13

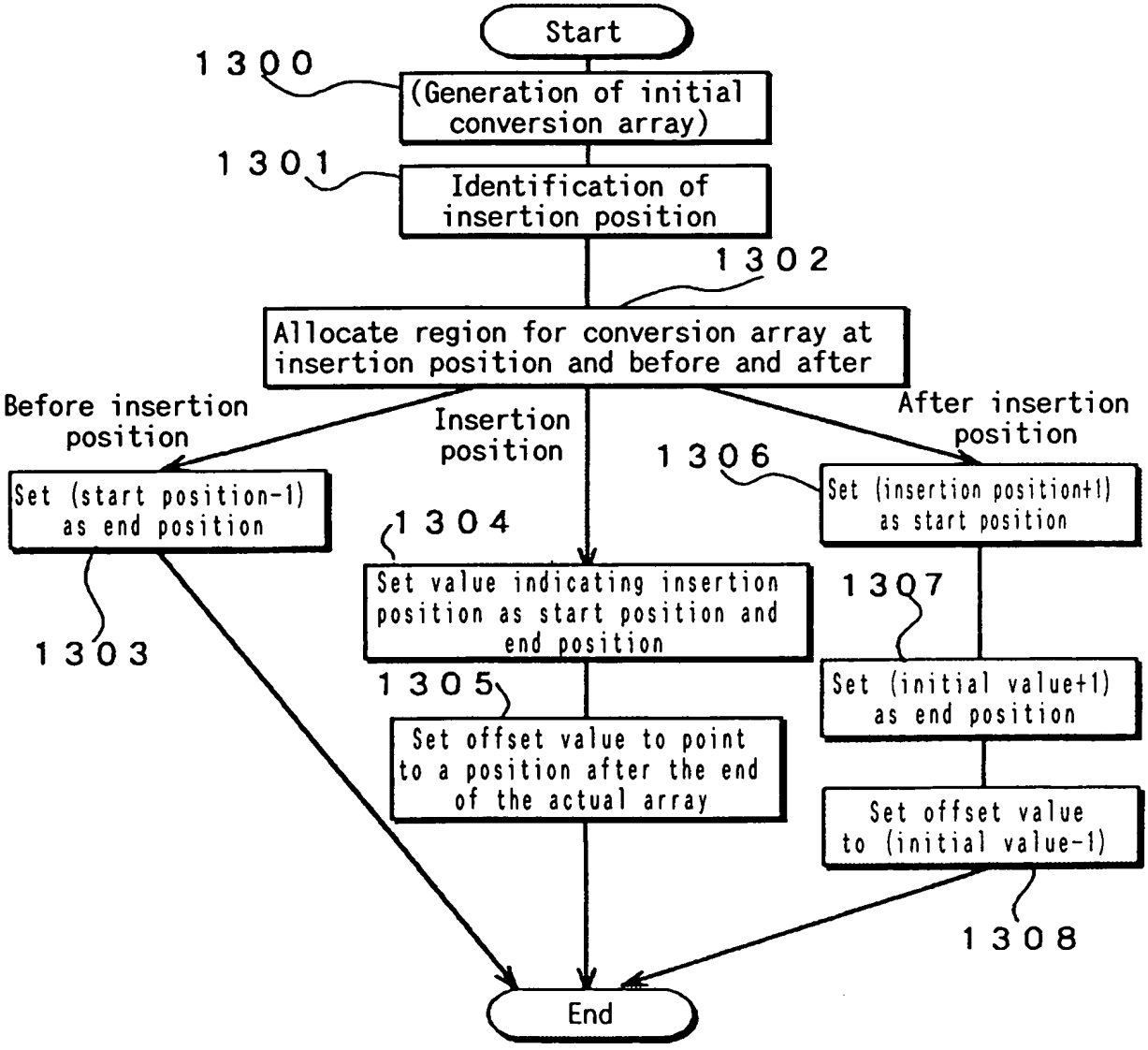

Start
1300
(Generation of initial conversion array)
1301
Identification of insertion position
1302
Allocate region for conversion array at insertion position and before and after
Before insertion position
Insertion position
After insertion position
1306
Set (start position-1) as end position
1304
Set (insertion position+1) as start position
1303
Set value indicating insertion position as start position and end position
1307
1305
Set (initial value+1) as end position
Set offset value to point to a position after the end of the actual array
Set offset value to (initial value-1)
1308
End

FIG. 14

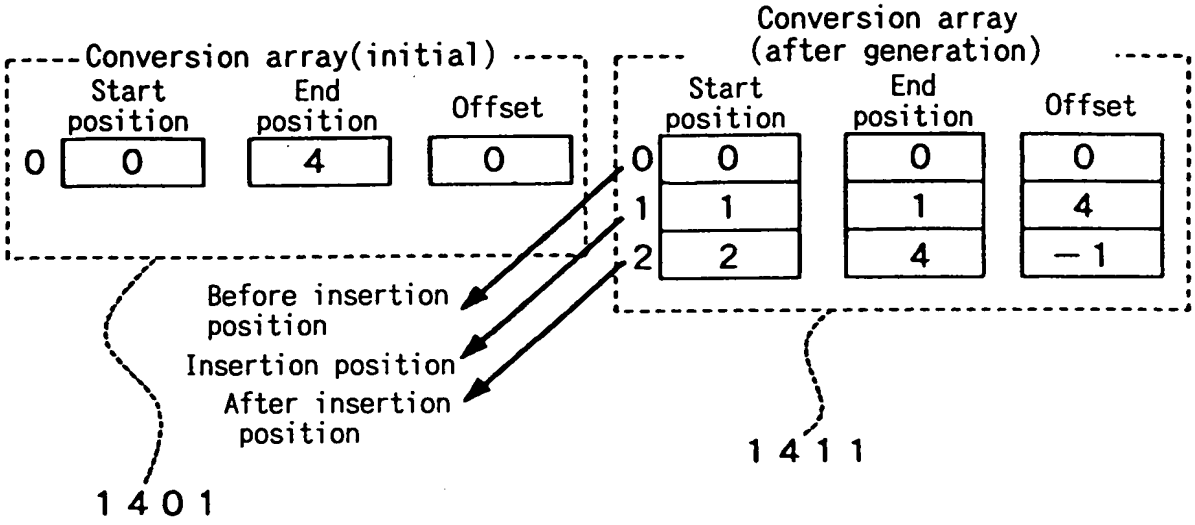
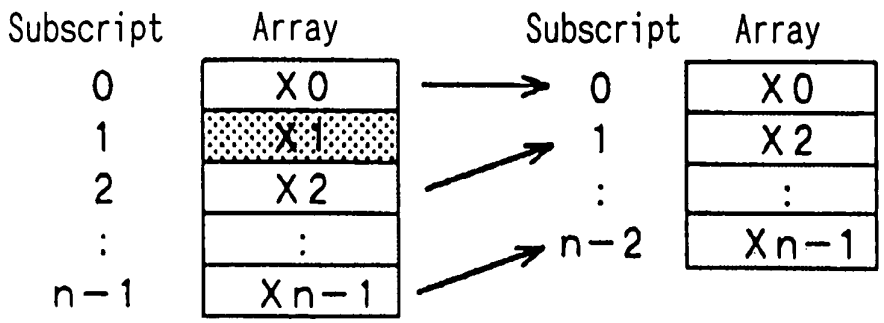

Conversion array(initial)
Start position
End position
Offset
0
0
4
0
Conversion array (after generation)
Start position
End position
Offset
0
0
0
0
1
1
1
4
2
2
4
−1
Before insertion position
Insertion position
After insertion position
1401
1411

FIG. 15A



Deletion of an element from an array of size "n"

FIG. 15B

Subscript	Stored value
0	X0
1	X2
2	X3
:	:
:	:
n−2	Xn−1

Logical relationship between subscripts and stored values

FIG. 15C

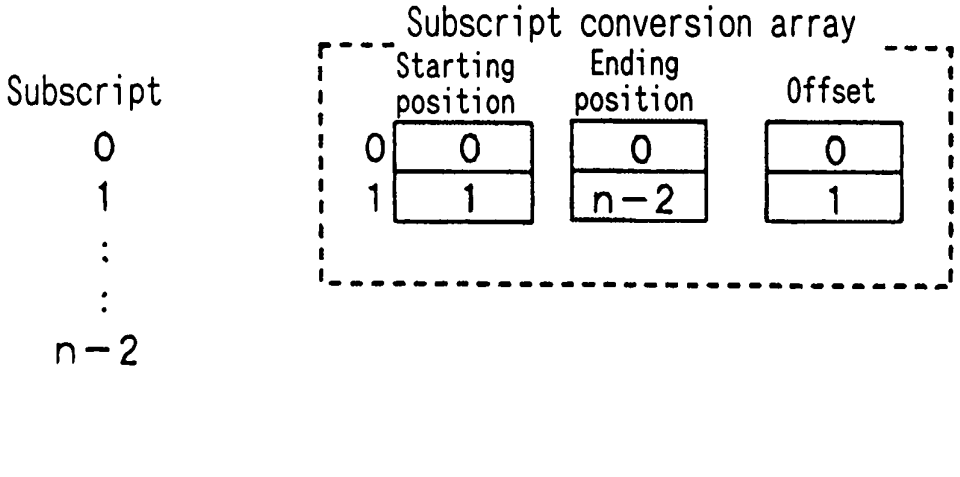


FIG. 16A

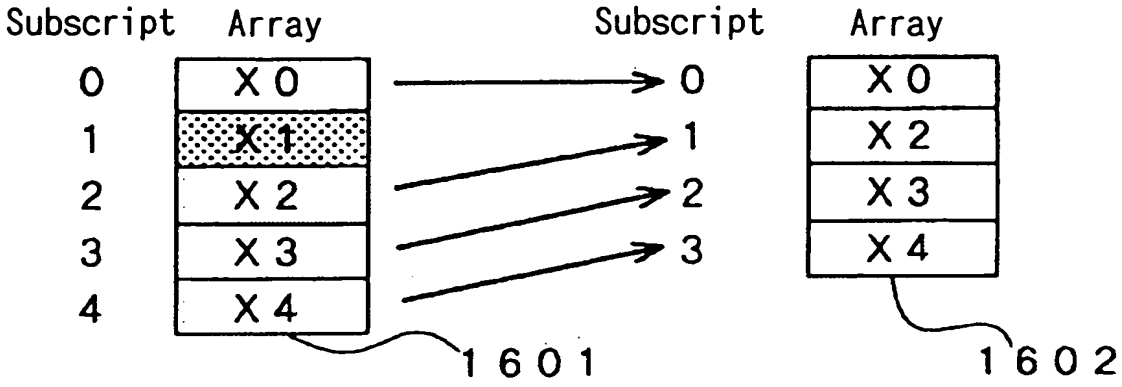
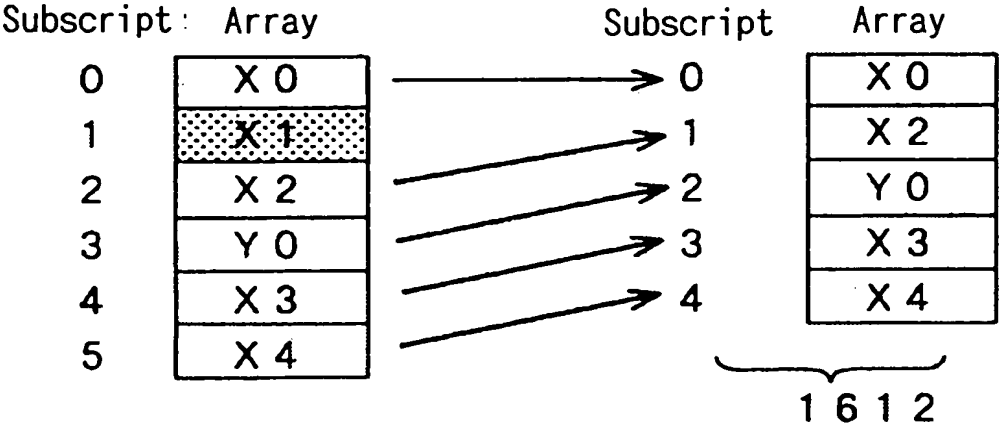
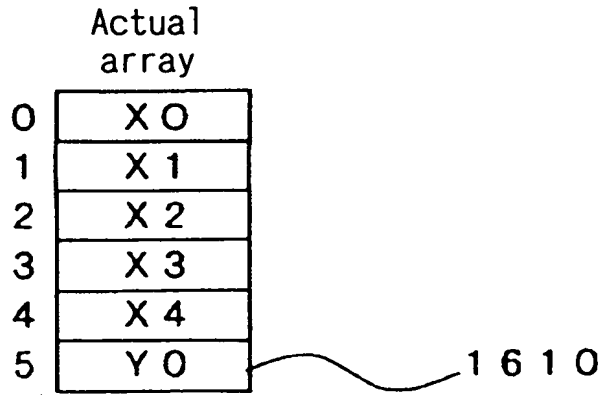


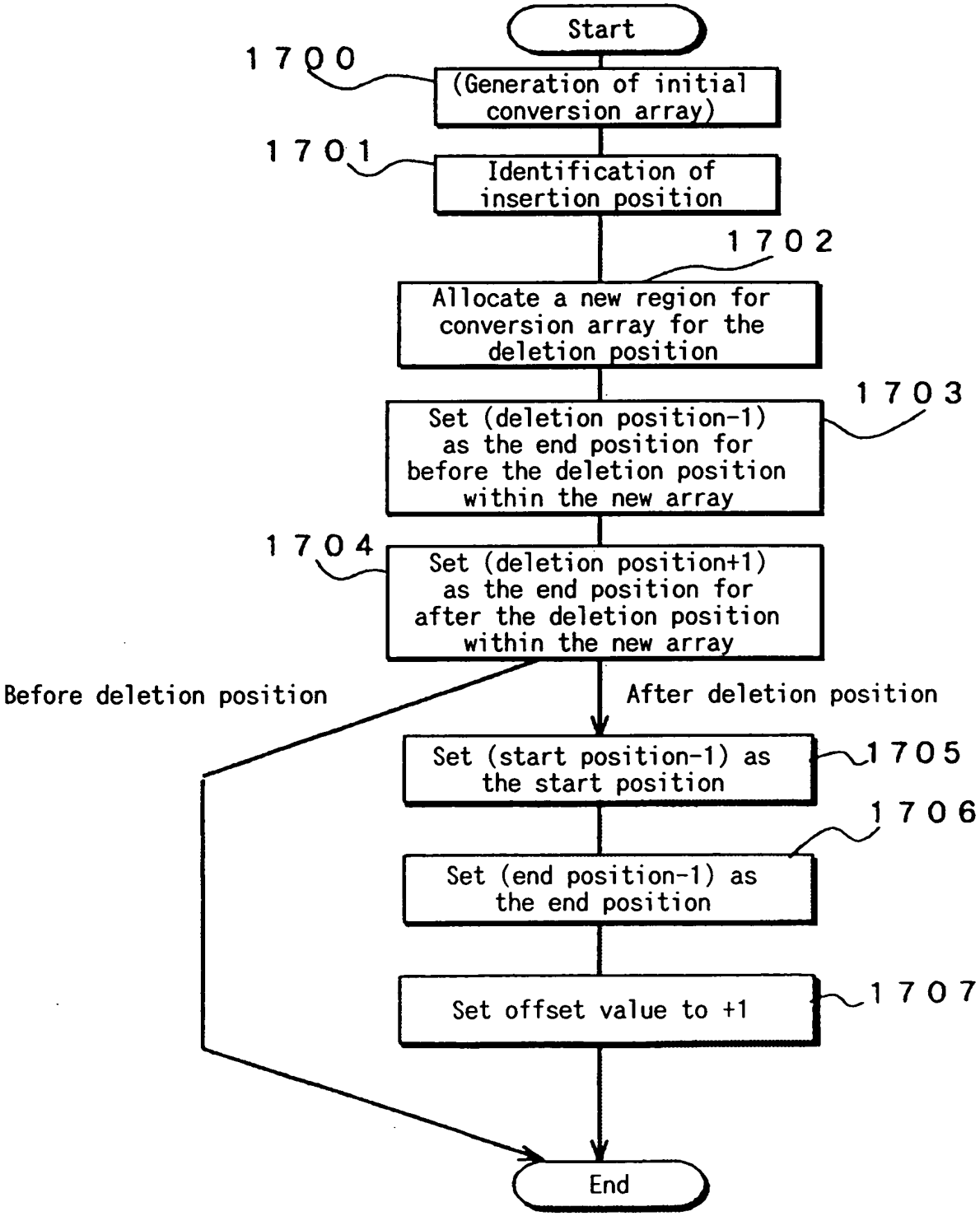
FIG. 16B





JP020048231

FIG. 17


Start
1700
(Generation of initial conversion array)
1701
Identification of insertion position
1702
Allocate a new region for conversion array for the deletion position
1703
Set (deletion position-1) as the end position for before the deletion position within the new array
1704
Set (deletion position+1) as the end position for after the deletion position within the new array
Before deletion position
After deletion position
Set (start position-1) as the start position
1705
1706
Set (end position-1) as the end position
Set offset value to +1
1707
End

10/048231

FIG. 18A

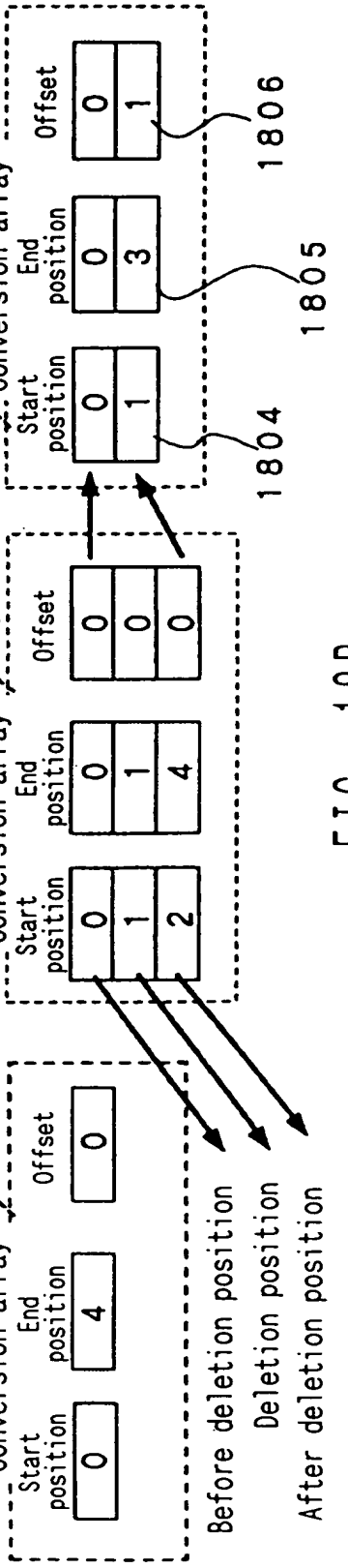
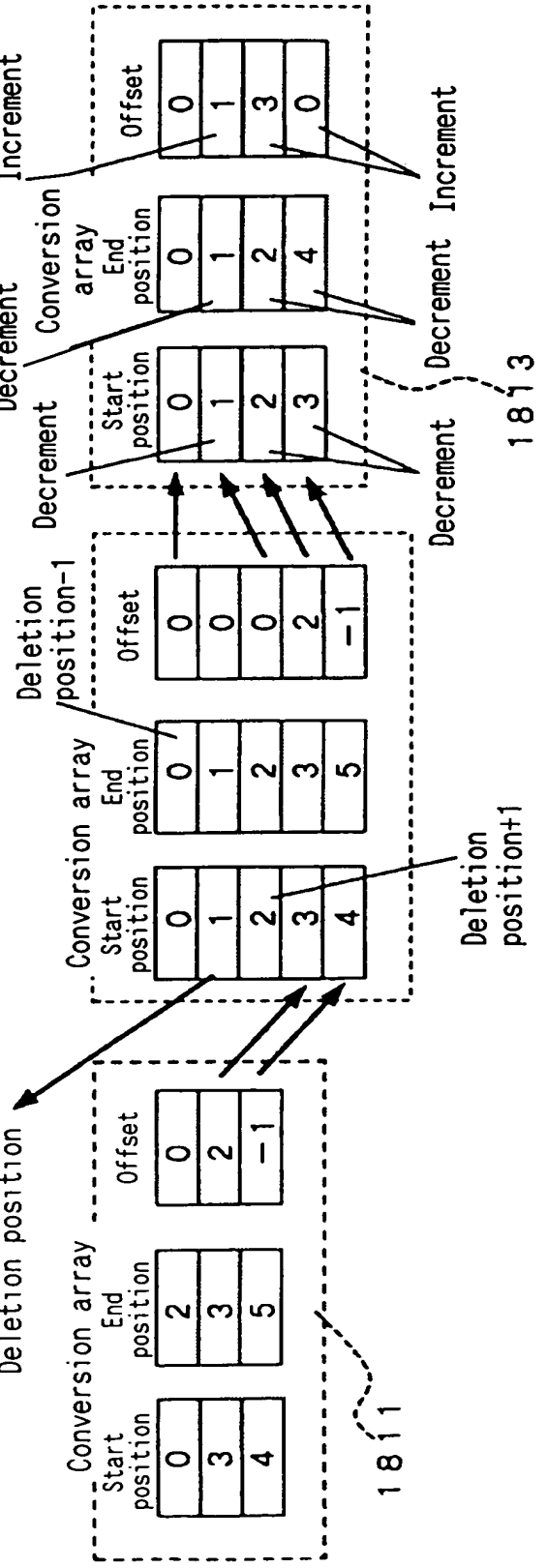


FIG. 18B



2002-12-26T00:00:00

FIG. 19

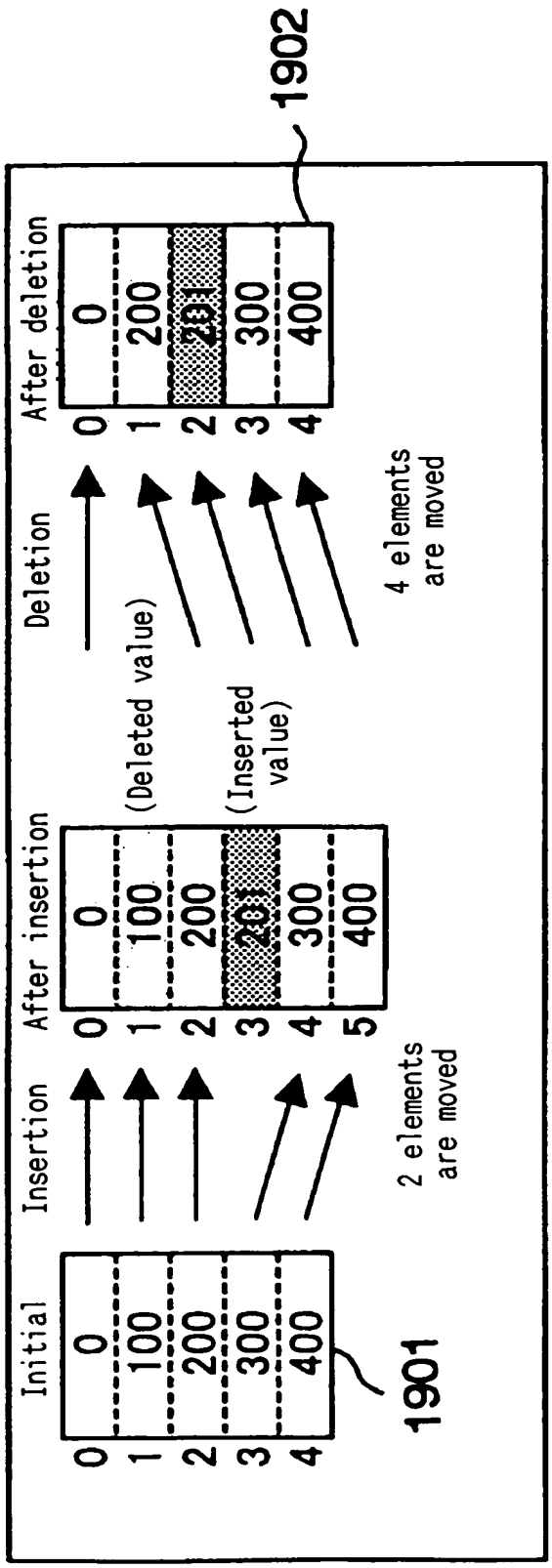


FIG. 20

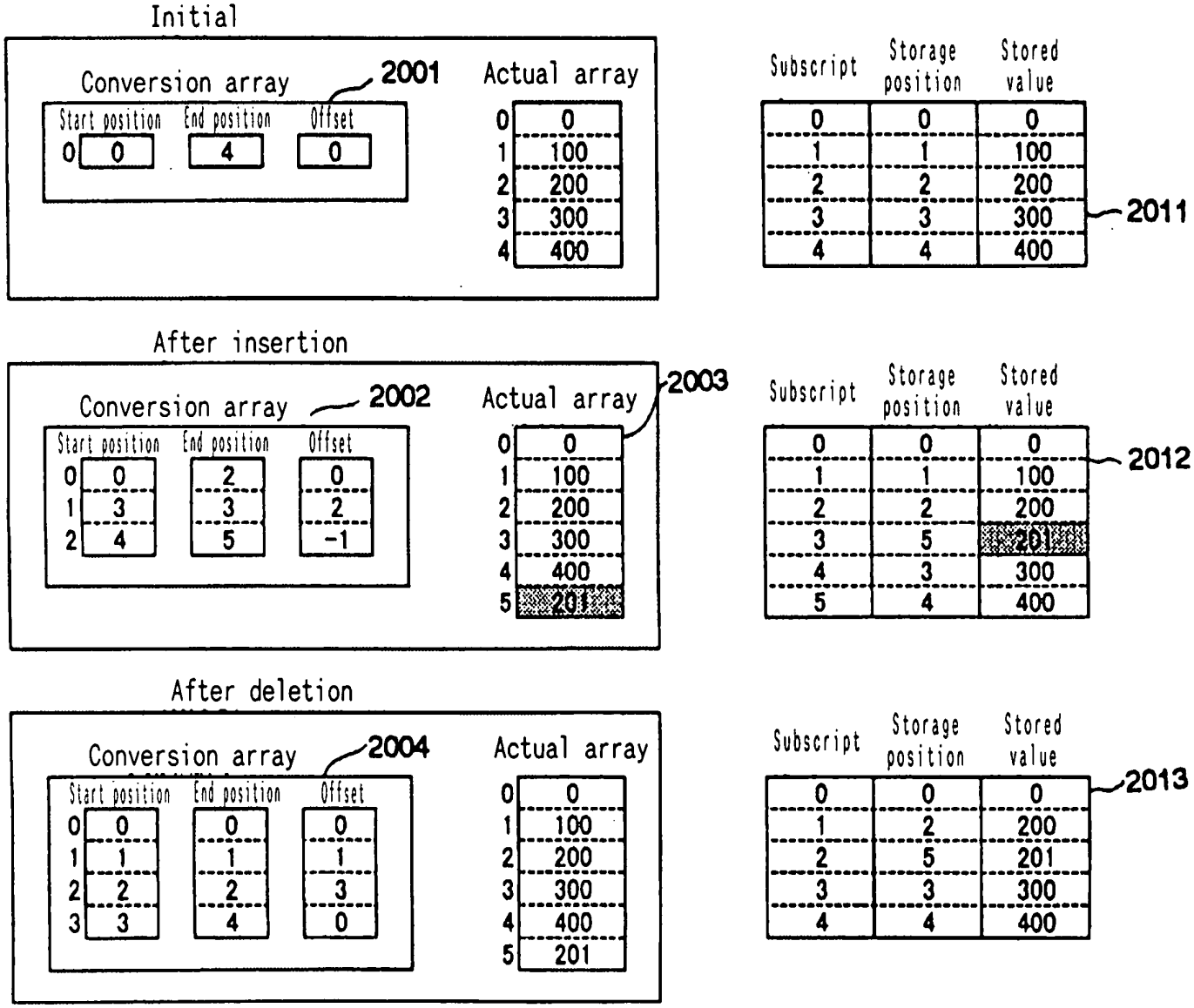

Initial
Conversion array
2001
Start position
End position
Offset
0 0 4 0
Actual array
0 0
1 100
2 200
3 300
4 400
Subscript
Storage position
Stored value
0 0 0
1 1 100
2 2 200
3 3 300
4 4 400
2011
After insertion
Conversion array
2002
Start position
End position
Offset
0 0 2 0
1 3 3 2
2 4 5 -1
Actual array
2003
0 0
1 100
2 200
3 300
4 400
5 201
Subscript
Storage position
Stored value
0 0 0
1 1 100
2 2 200
3 5 201
4 3 300
5 4 400
2012
After deletion
Conversion array
2004
Start position
End position
Offset
0 0 0 0
1 1 1 1
2 2 2 3
3 3 4 0
Actual array
0 0
1 100
2 200
3 300
4 400
5 201
Subscript
Storage position
Stored value
0 0 0
1 2 200
2 5 201
3 3 300
4 4 400
2013

FIG. 21A

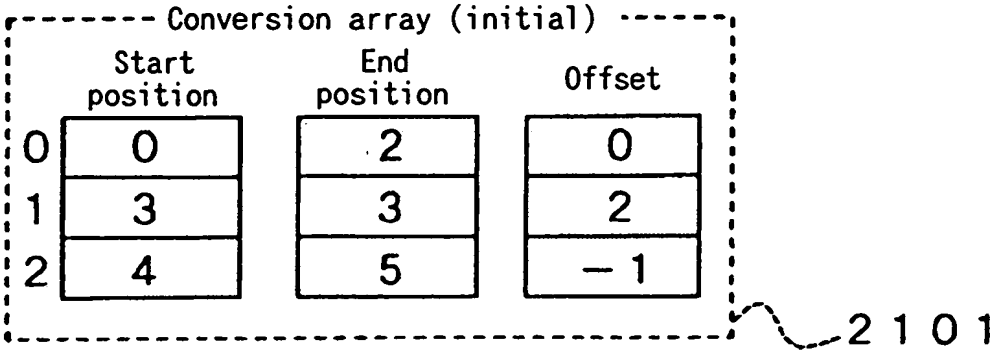
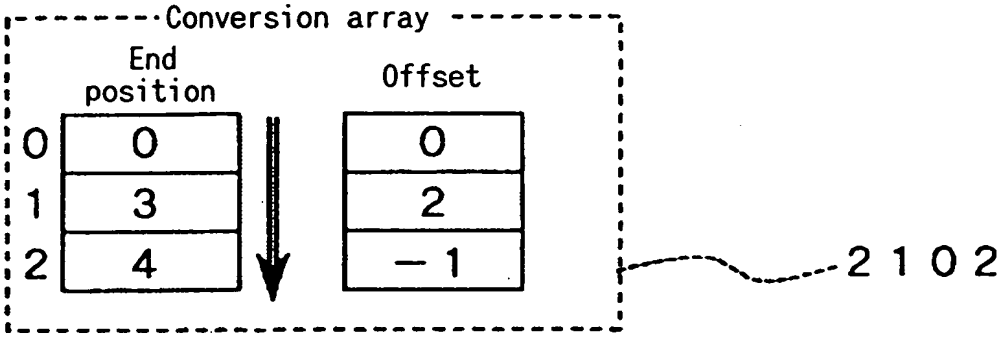
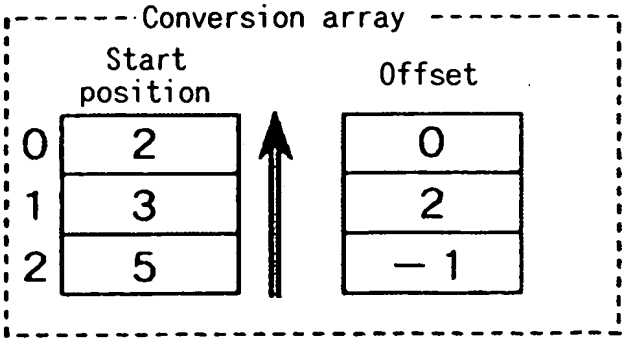


FIG. 21B



(End position only; logic: ≦)

FIG. 21C



(Start position only; logic: ≧)

FIG. 22

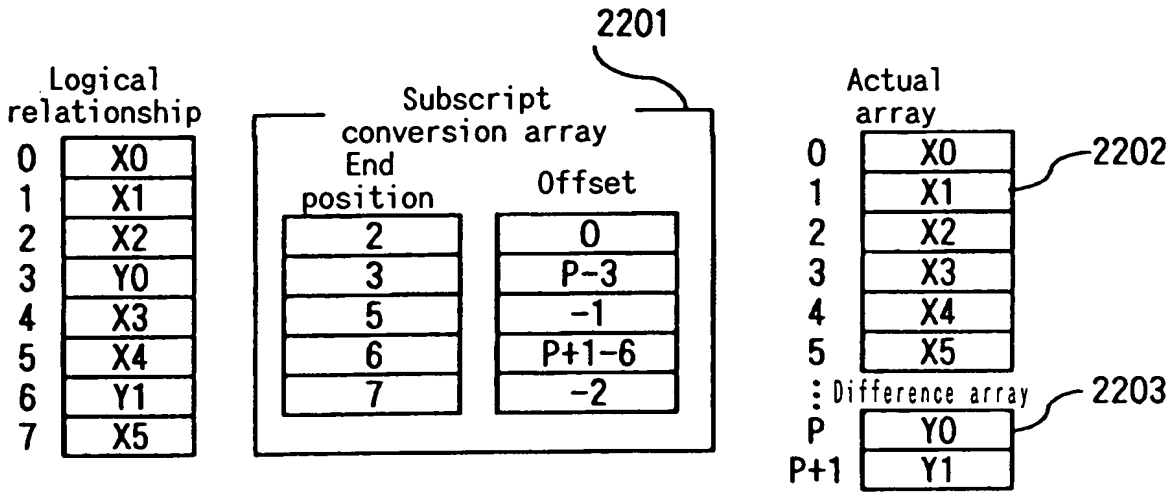
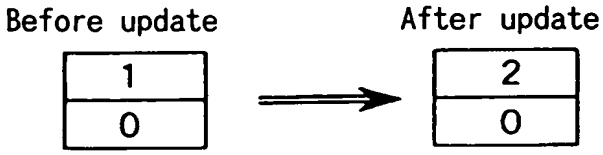


FIG. 23A



Increment values of 1 or more in the array.

FIG. 23B

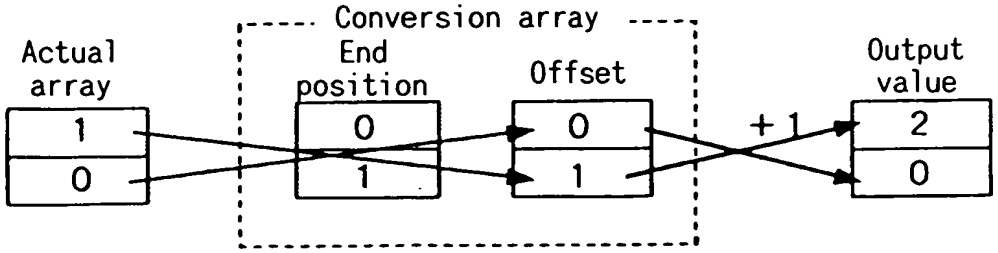
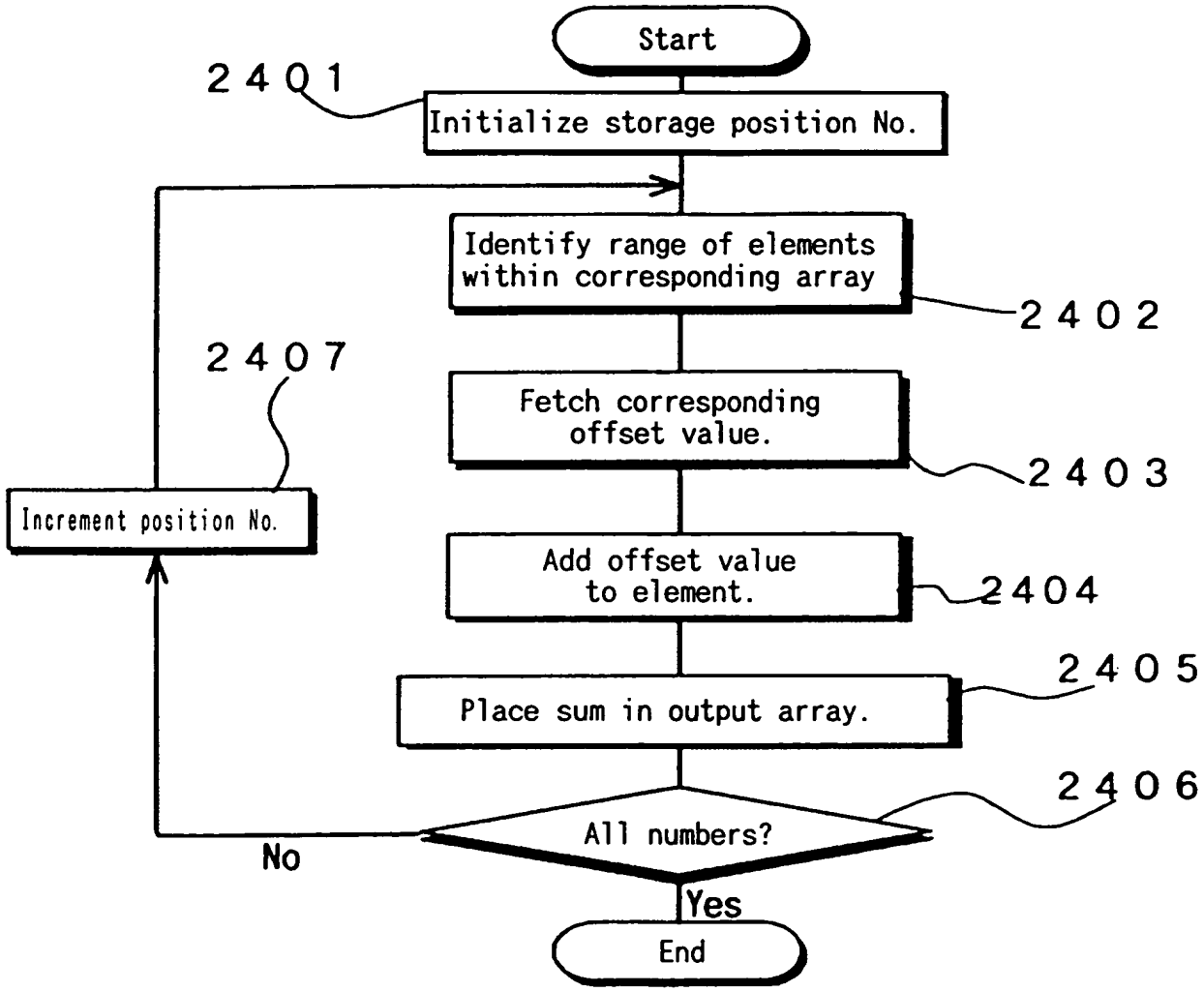
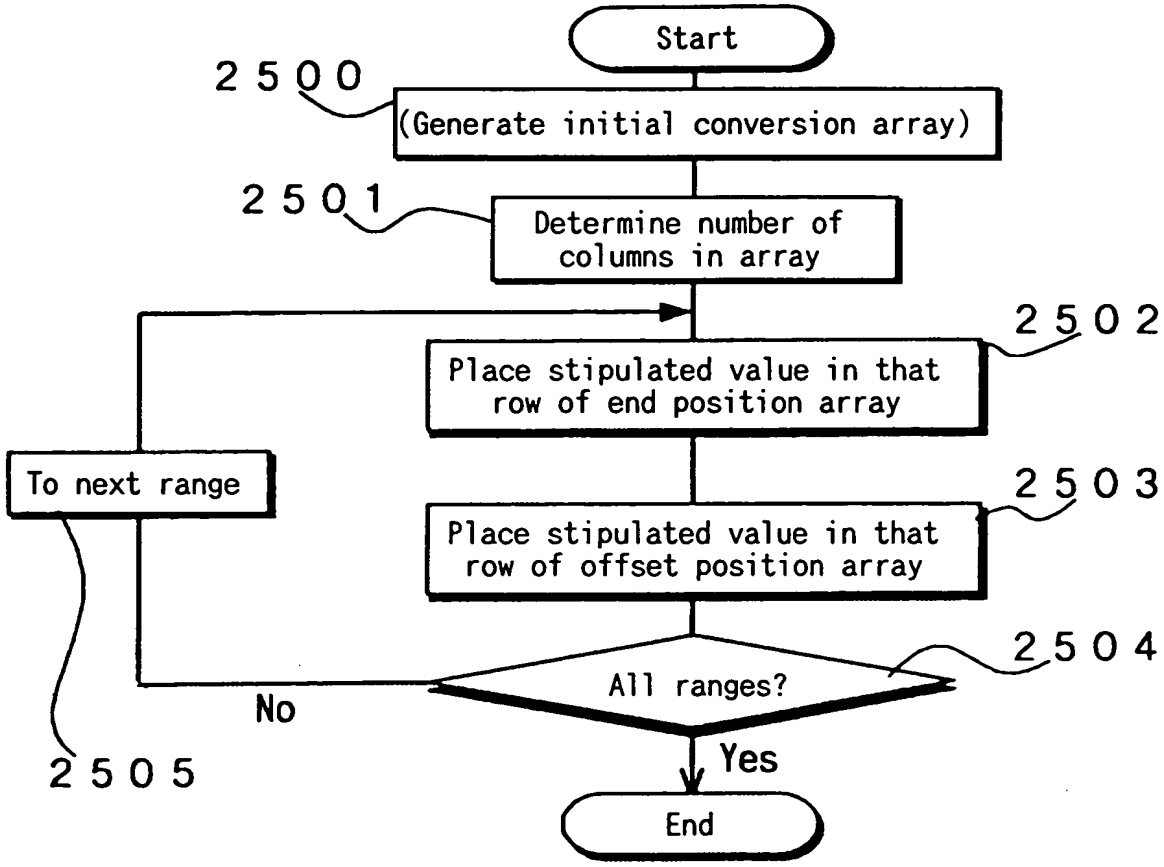


FIG. 24



FOR OFFICIAL USE ONLY

FIG. 25



FOR OFFICIAL USE ONLY

FIG. 26A

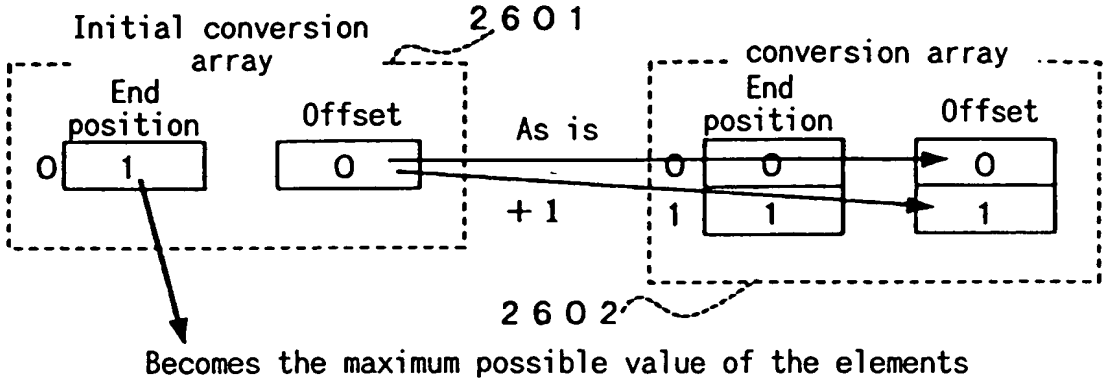
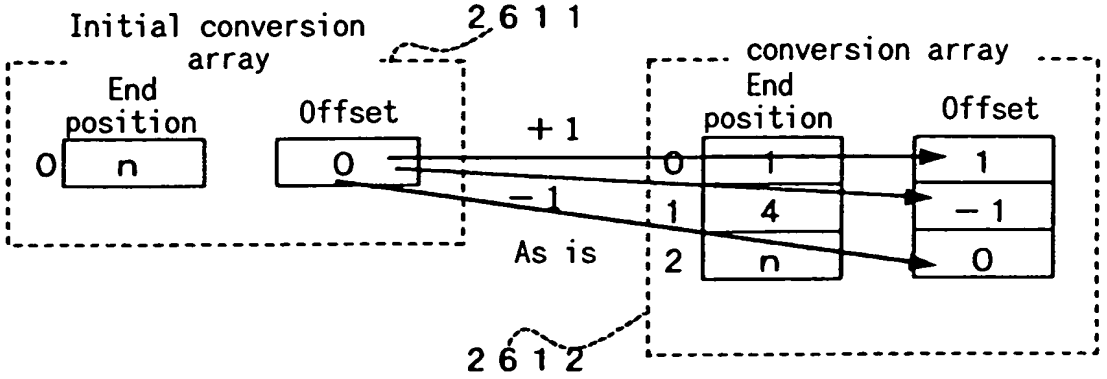


FIG. 26B



2003-12-03 14:08:08

FIG. 27

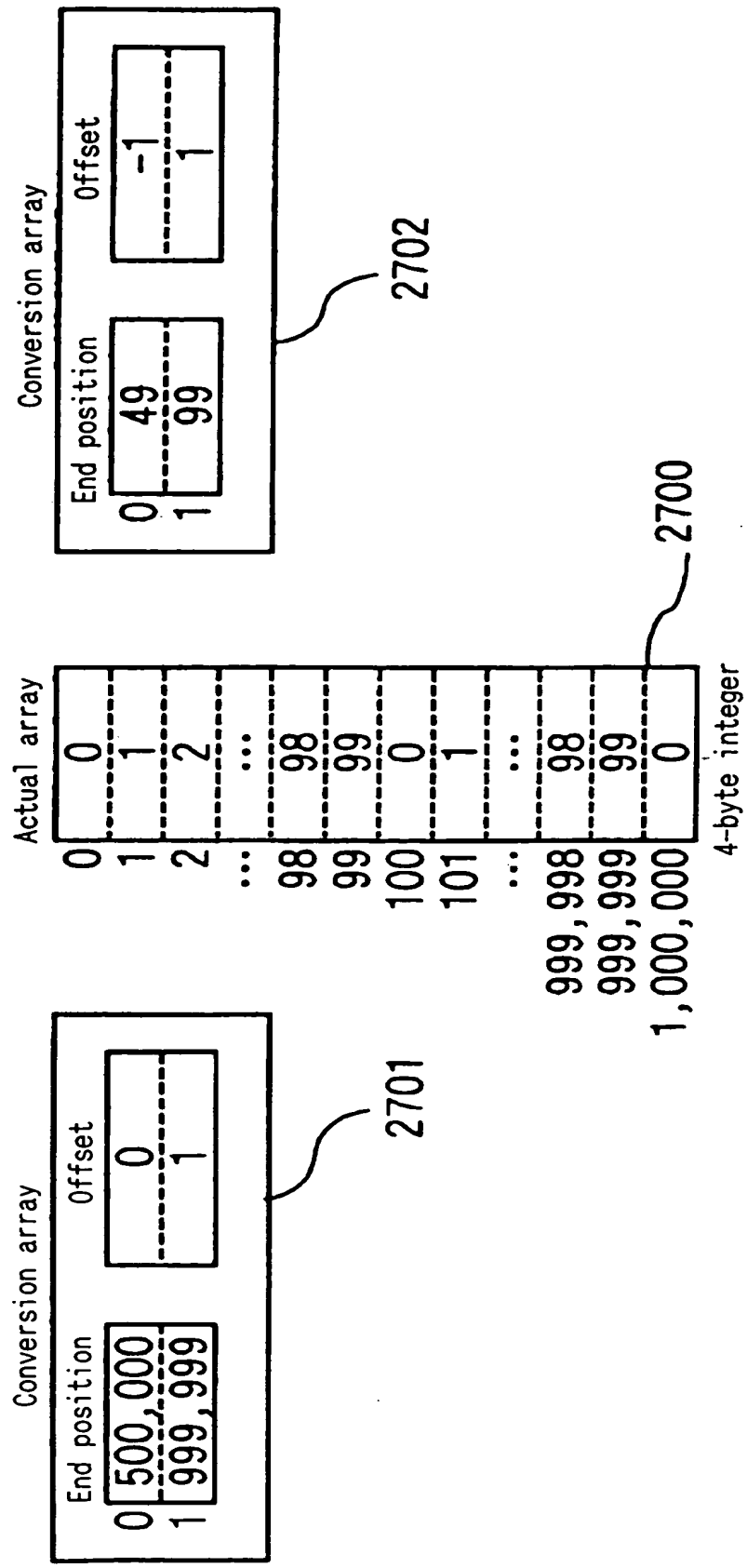

Conversion array
End position
Offset
0
1
49
99
-1
1
2702
Actual array
0
1
2
...
98
99
100
101
...
999,998
999,999
1,000,000
0
1
2
...
98
99
0
1
...
98
99
0
2700
4-byte integer
Conversion array
End position
Offset
0
1
500,000
999,999
0
1
2701

FIG. 28A

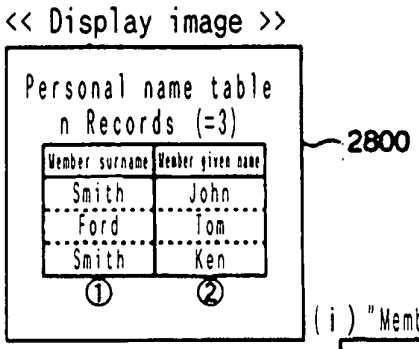


FIG. 28B

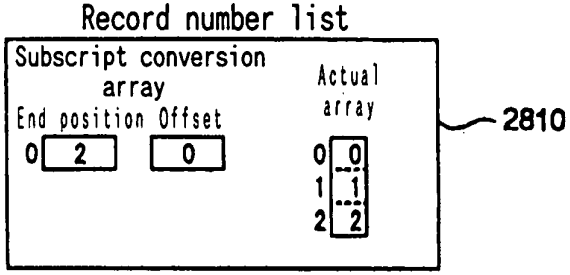


FIG. 28C

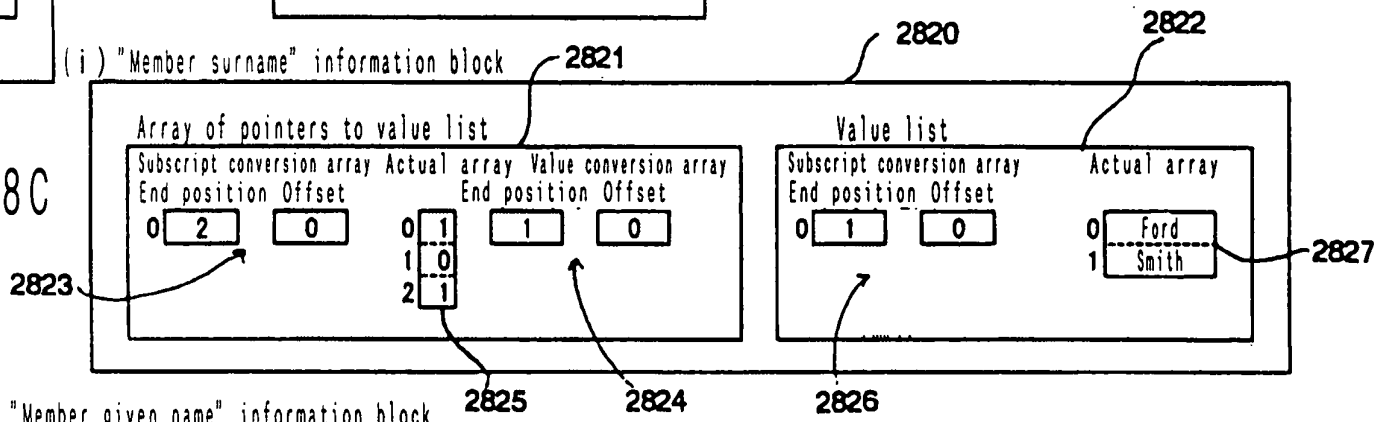


FIG. 28D

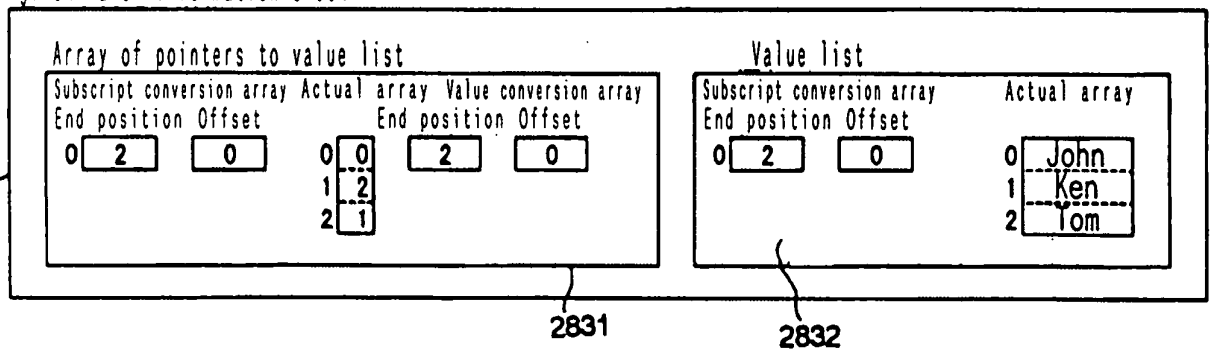


FIG. 29A

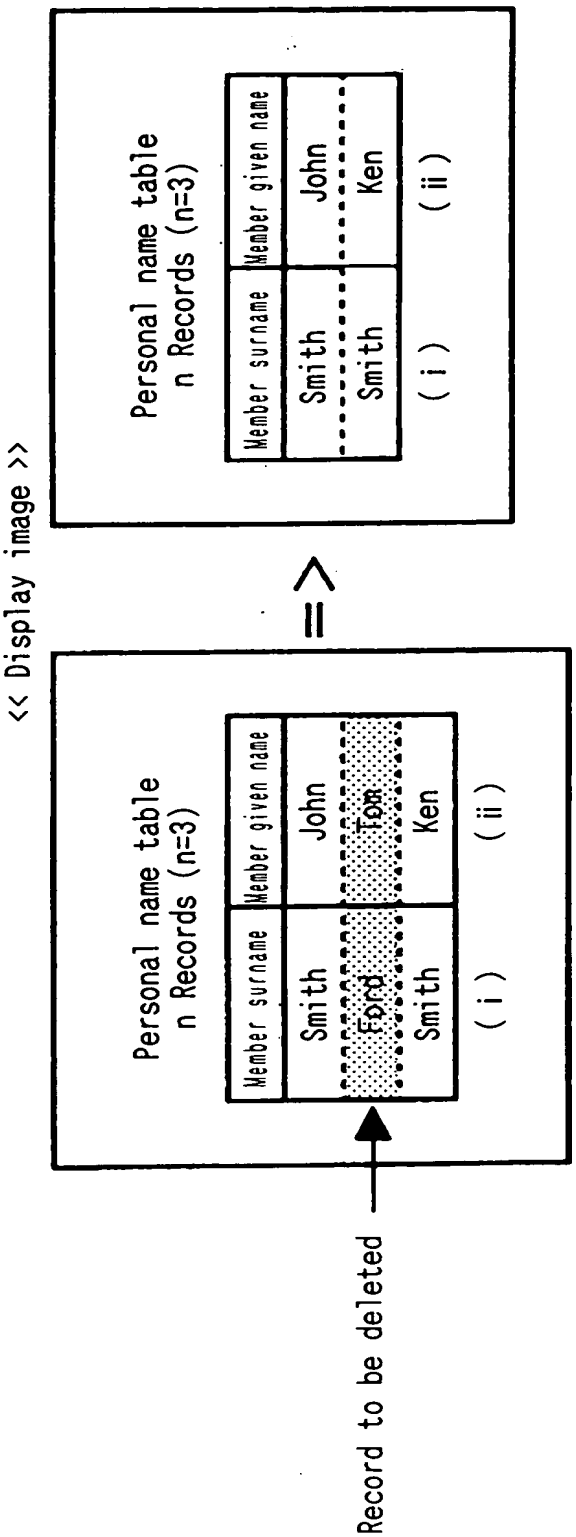
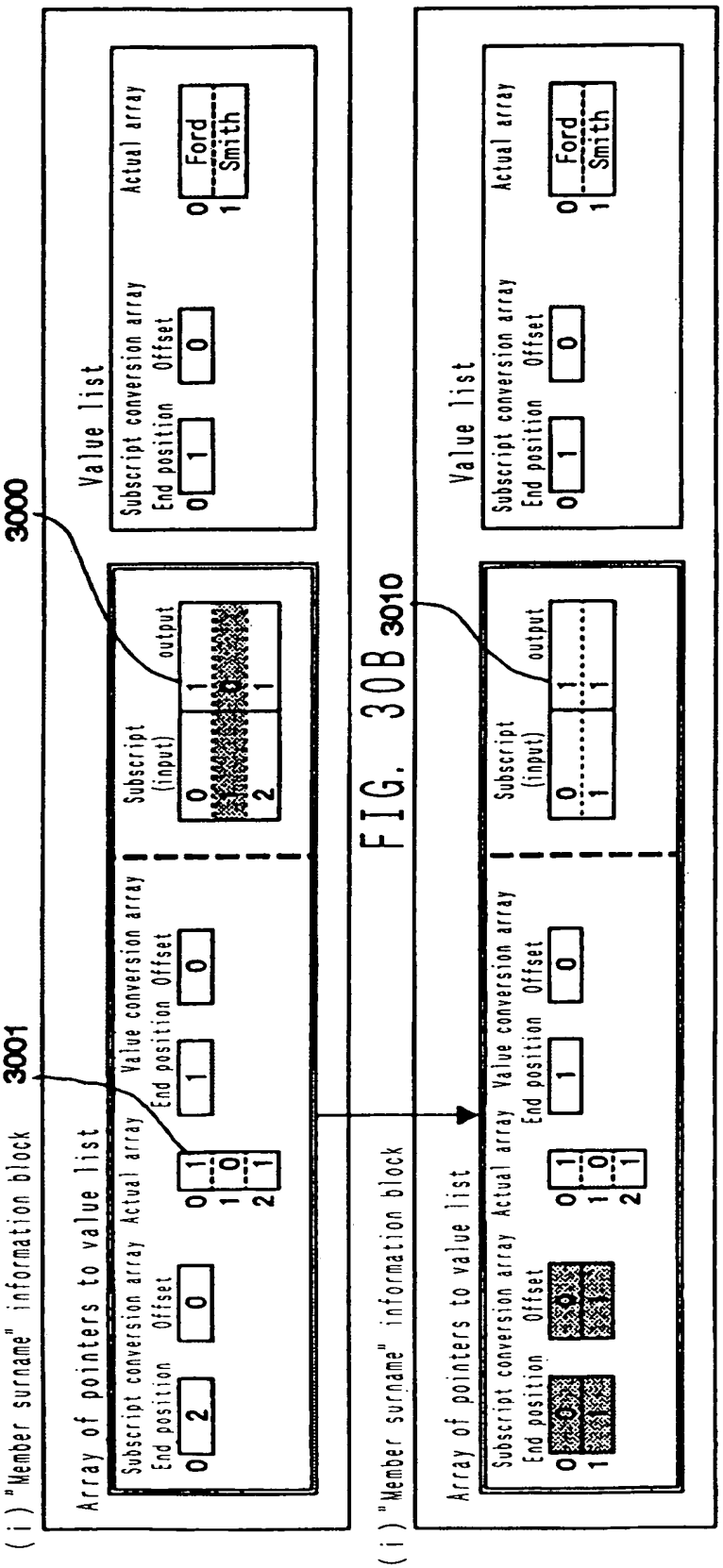
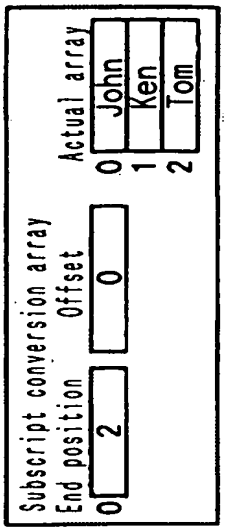


FIG. 29B

20020710 "Member surname" information block





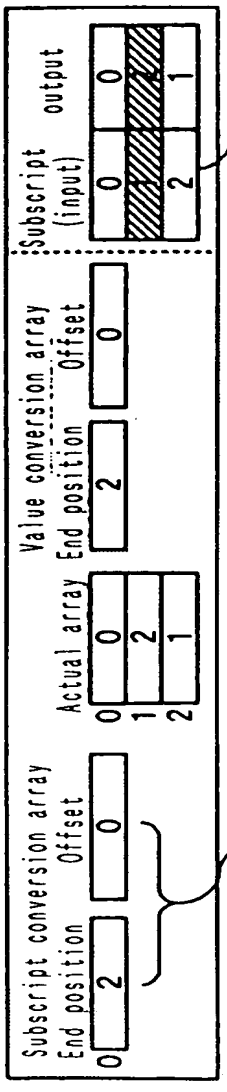
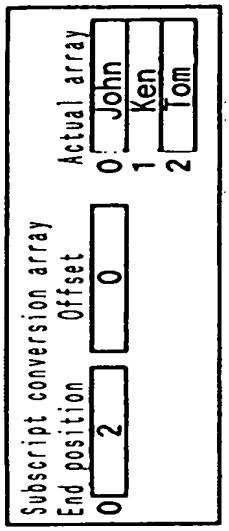


FIG. 31A



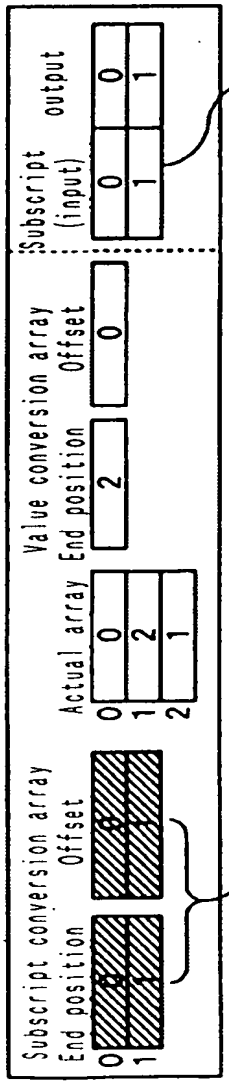
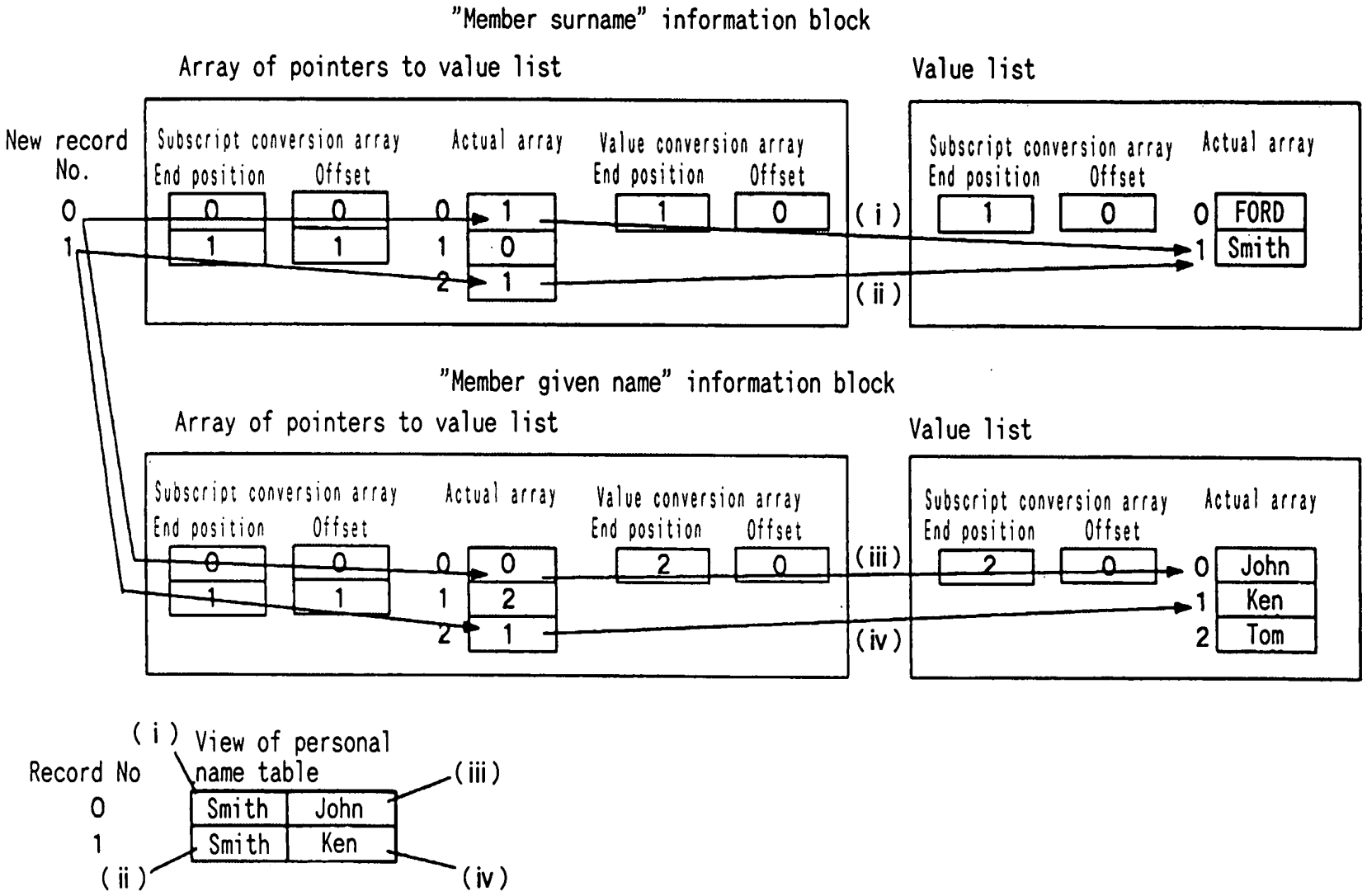


FIG. 31B

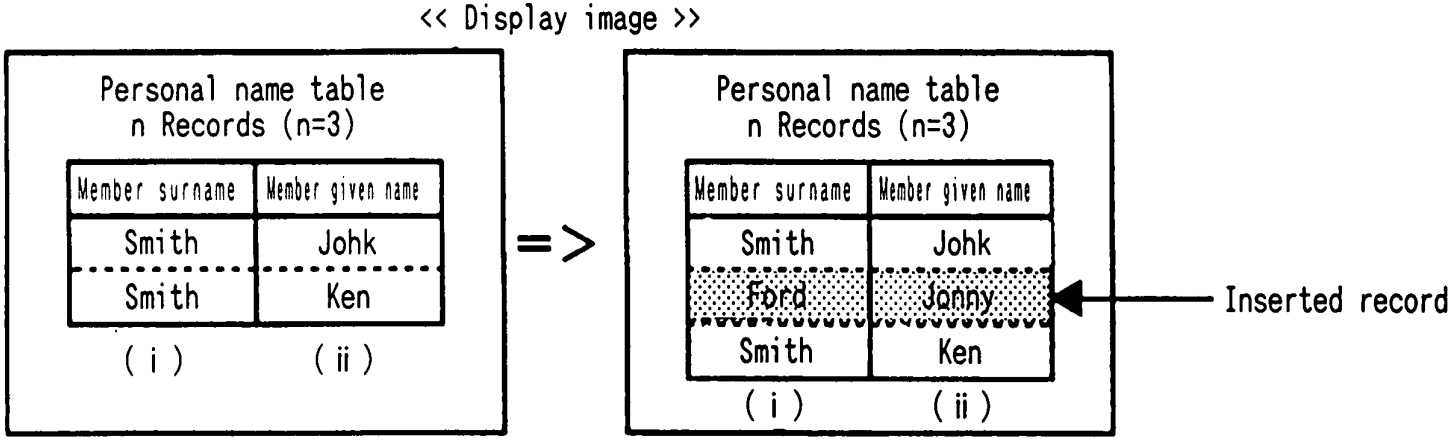
FIG. 32


"Member surname" information block
Array of pointers to value list
Value list
New record No.
Subscript conversion array
End position
Offset
Actual array
Value conversion array
End position
Offset
Subscript conversion array
End position
Offset
Actual array
0
1
0
1
0
1
2
1
0
1
0
1
0
1
0
1
0
FORD
1
Smith
(i)
(ii)
"Member given name" information block
Array of pointers to value list
Value list
Subscript conversion array
End position
Offset
Actual array
Value conversion array
End position
Offset
Subscript conversion array
End position
Offset
Actual array
0
1
0
1
2
1
0
1
0
1
0
1
0
2
0
2
0
John
1
Ken
2
Tom
(iii)
(iv)
(i) View of personal name table
Record No
0
1
Smith
John
Smith
Ken
(iii)
(ii)
(iv)

20020° 1628h00t

FIG. 33A

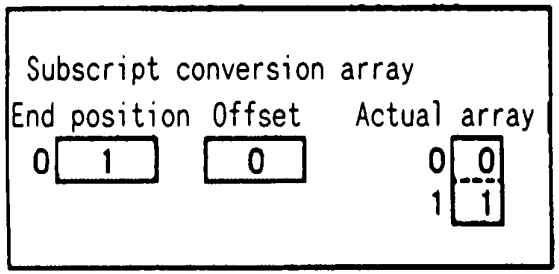
FIG. 33B



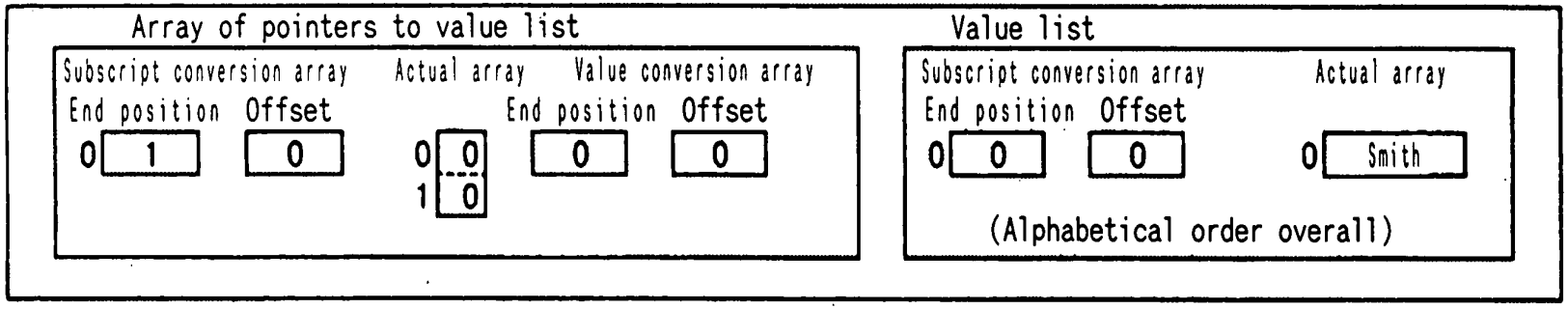
10/048231

FIG. 34

Record number list (before insertion)



(i) "Member surname" information block (before insertion)



(ii) "Member given name" information block (before insertion)

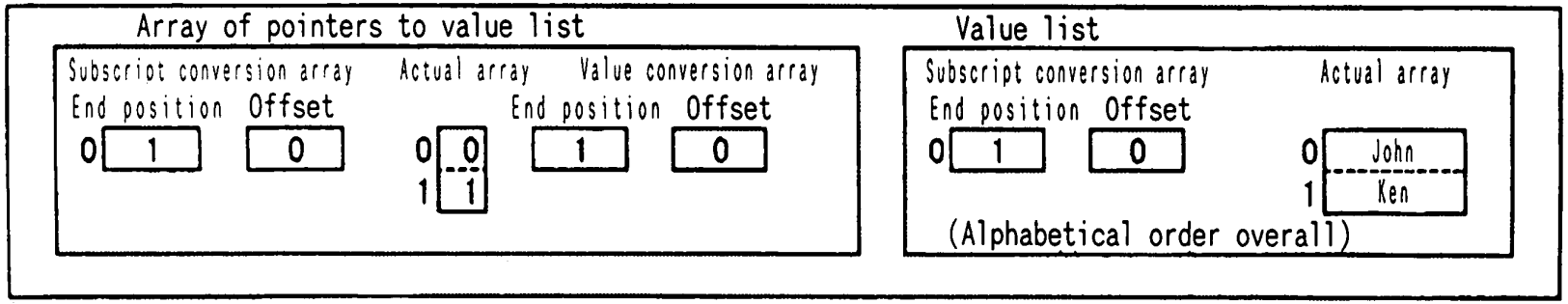


FIG. 35A

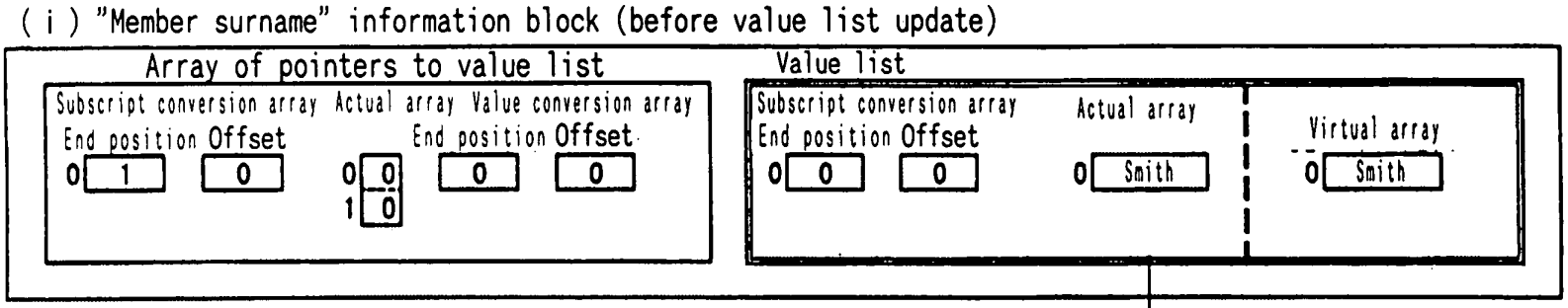


FIG. 35B

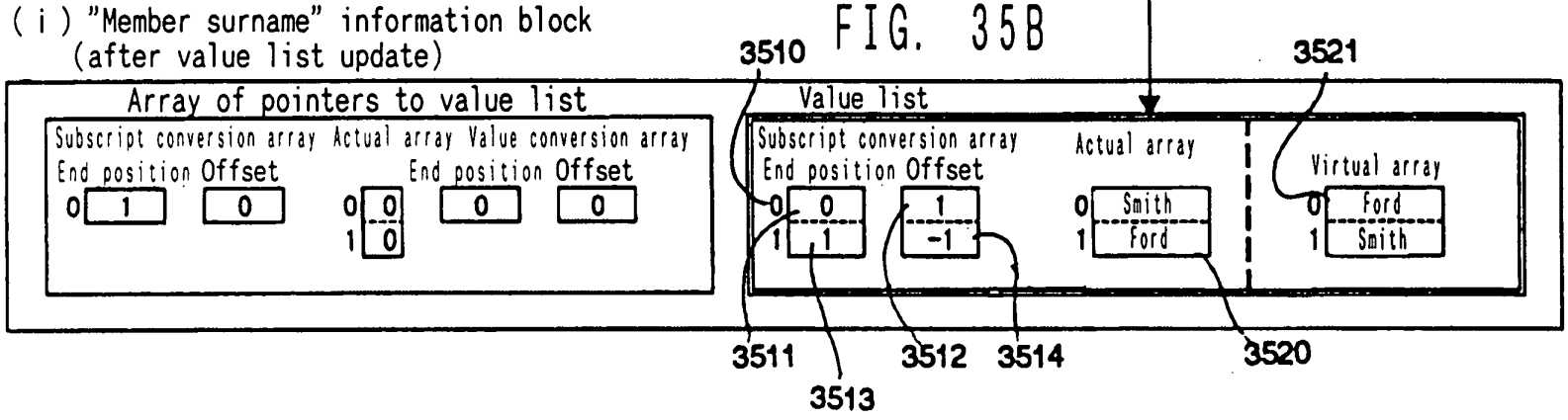


FIG. 36A

(ii) "Member given name" information block (before value list update)

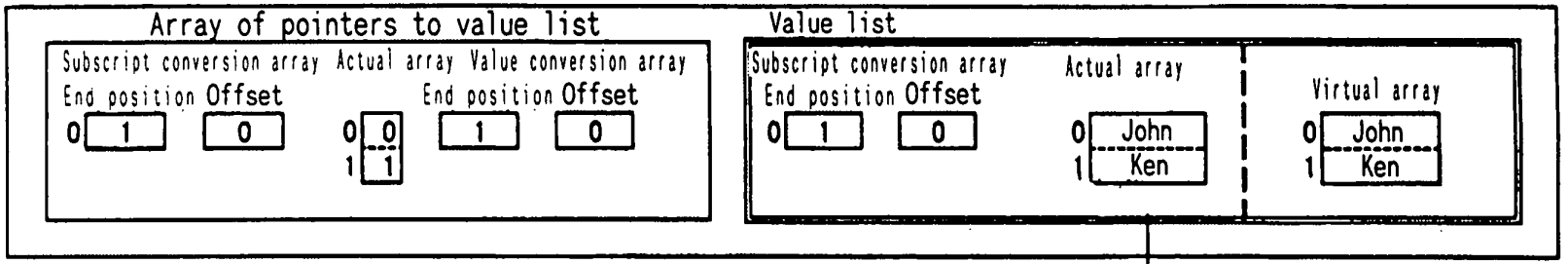
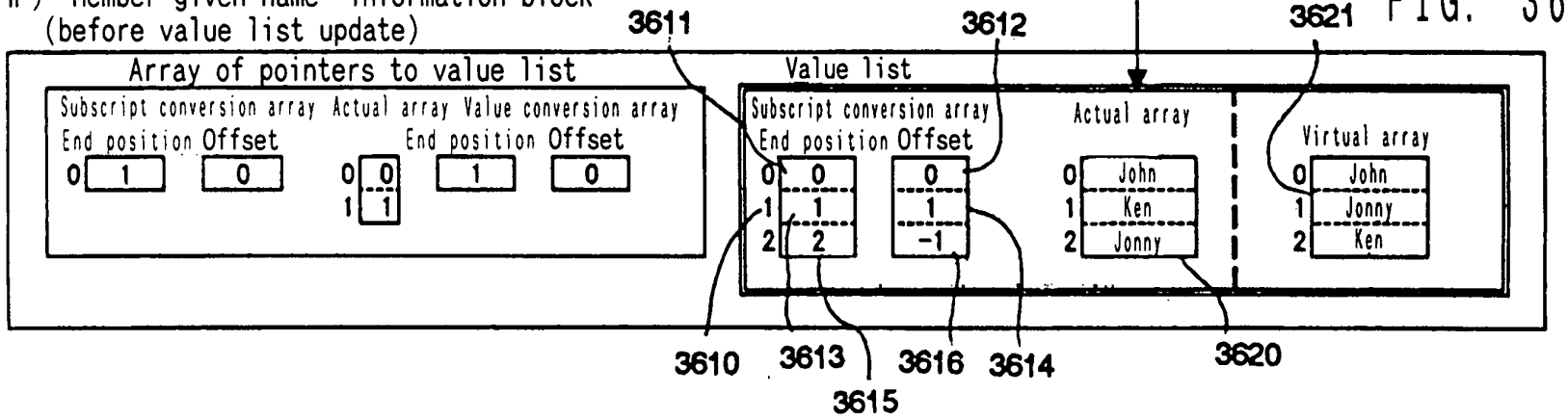


FIG. 36B

(ii) "Member given name" information block (before value list update)



20020729.1E2384007

FIG. 37A

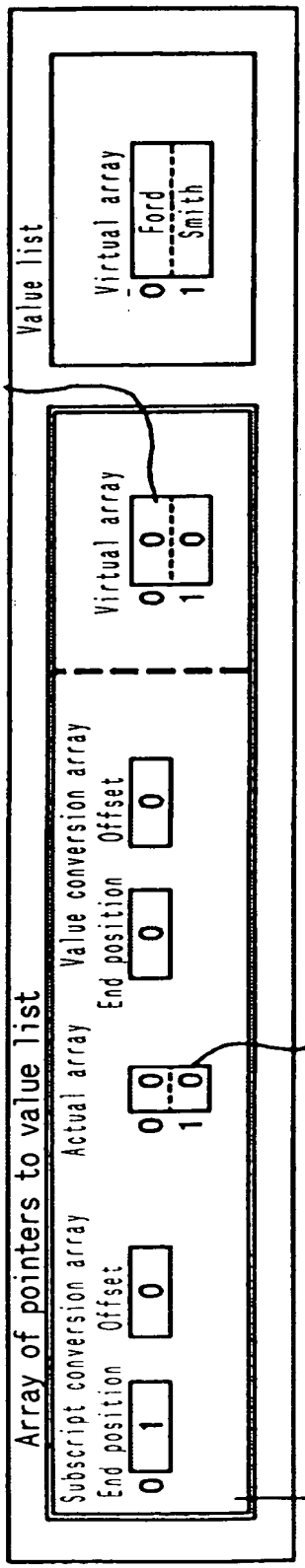


FIG. 37B

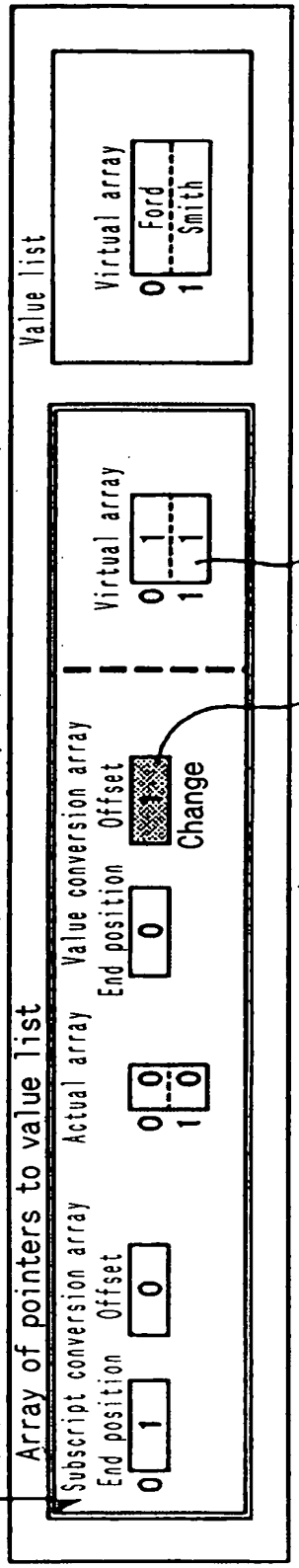


FIG. 38A

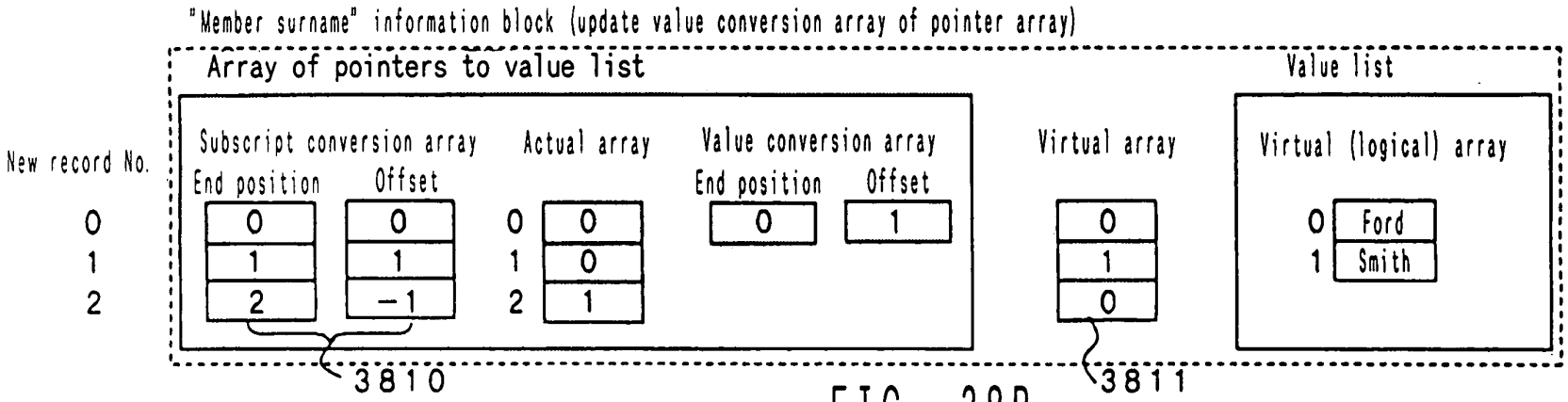


FIG. 38B

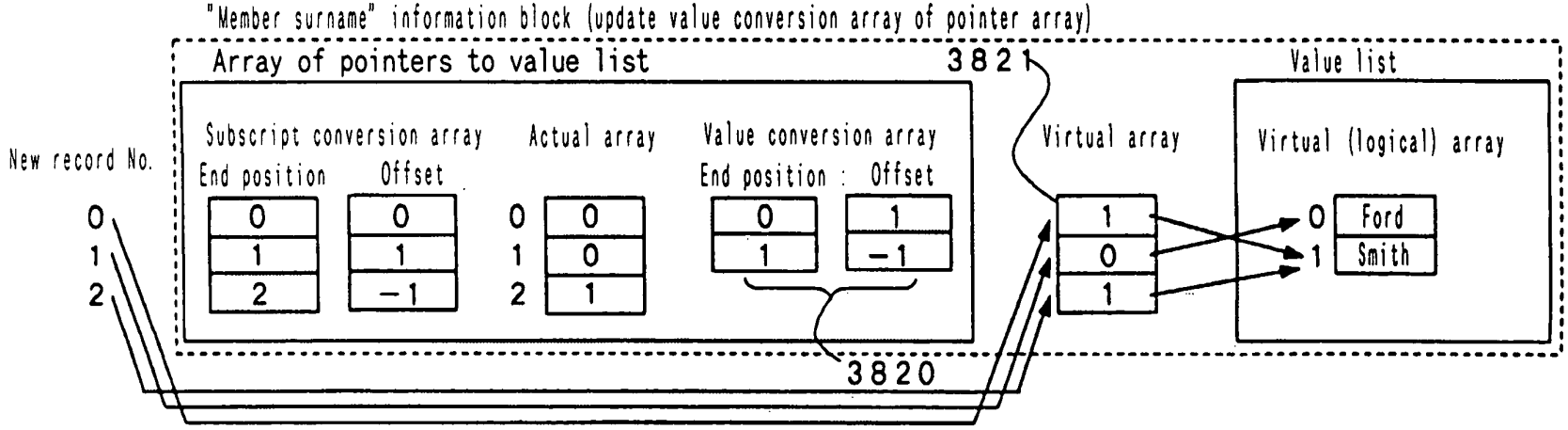


FIG. 39A

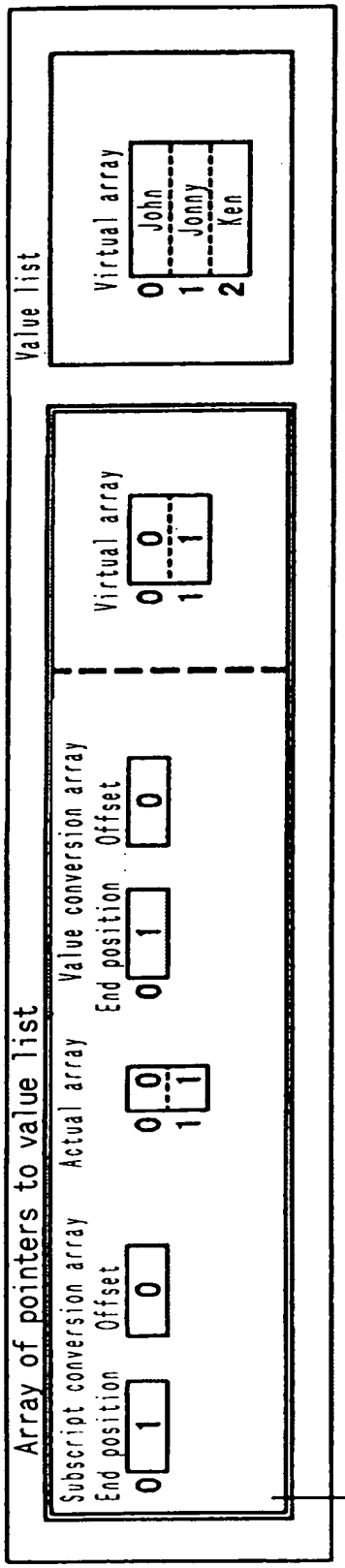


FIG. 39B

(ii) "Member given name" information block (after completion of update 1 of array of pointers to value list)

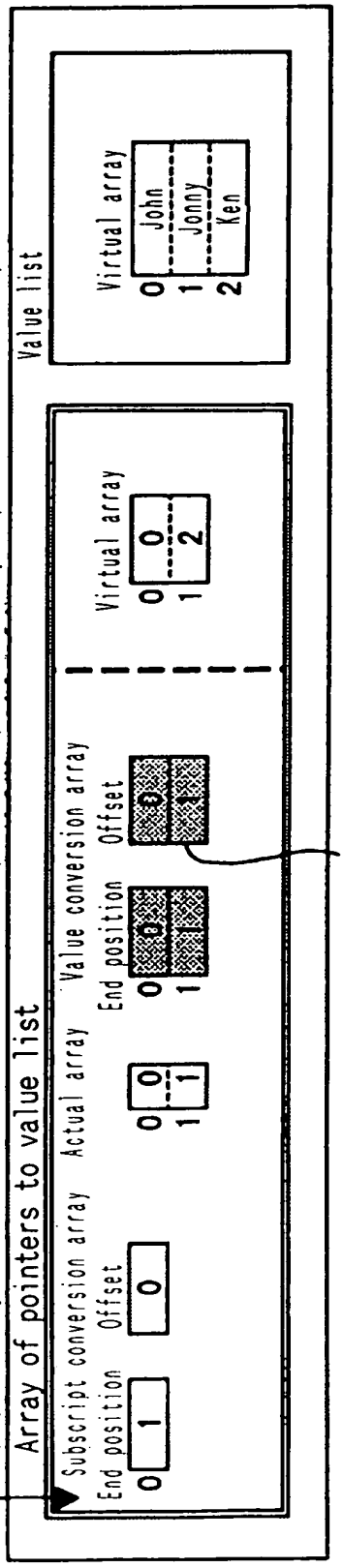


FIG. 40A

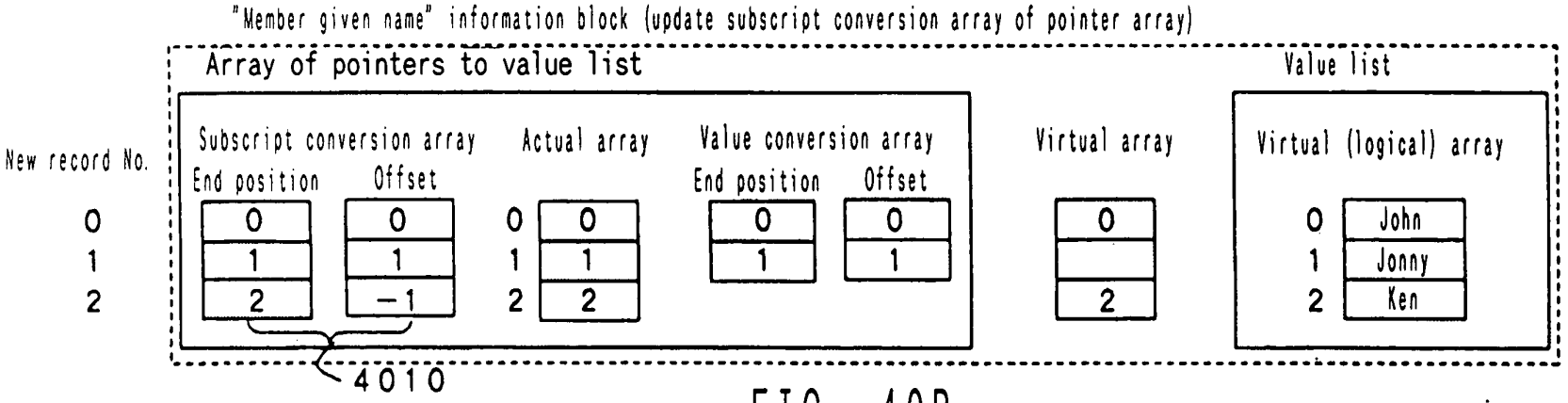


FIG. 40B

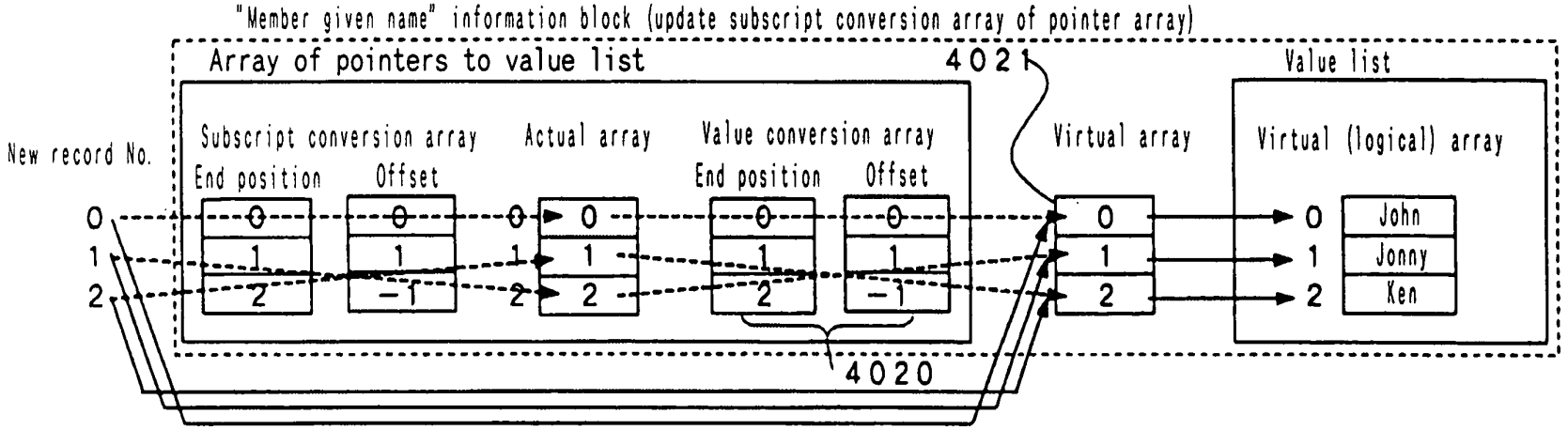


FIG. 41

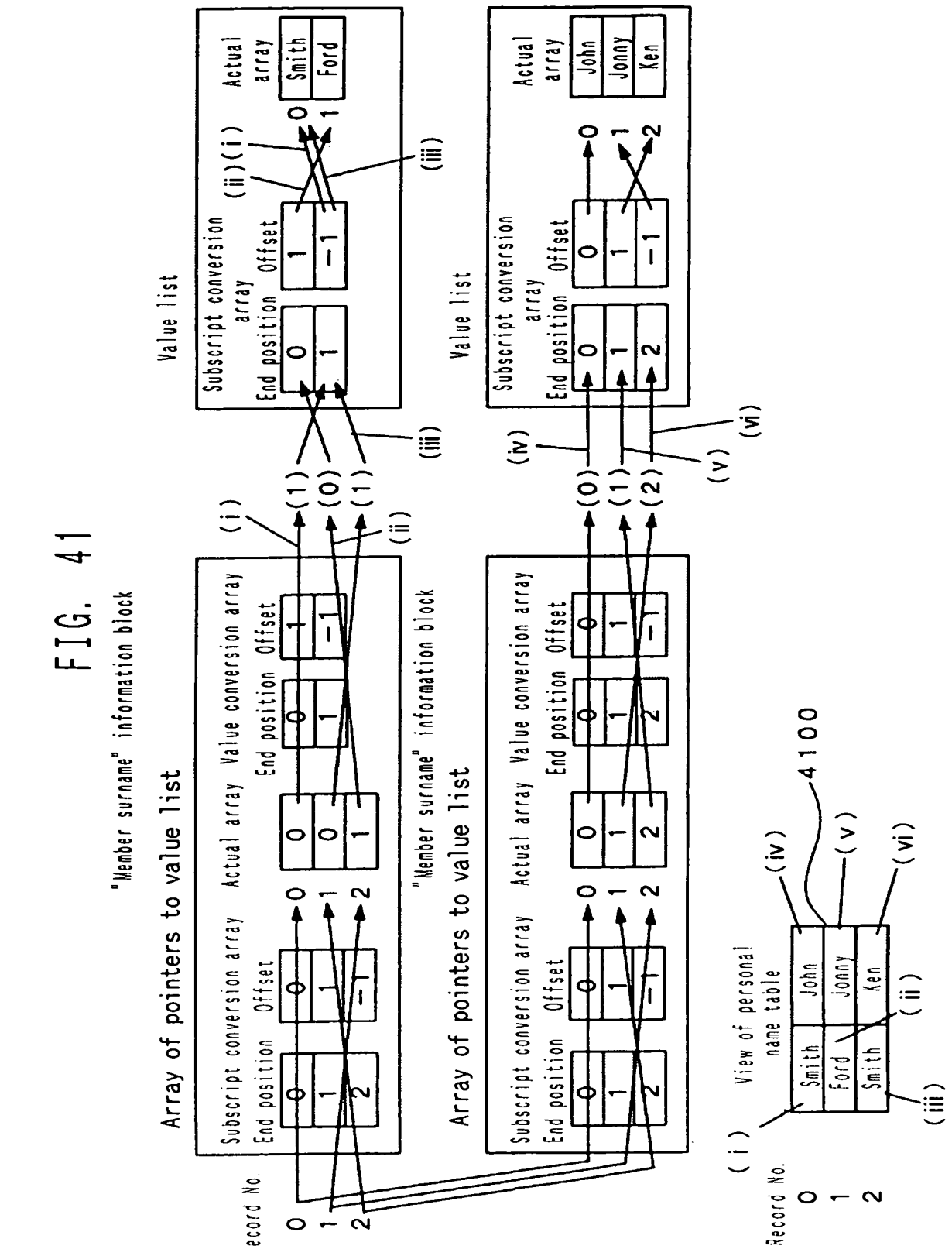

"Member surname" information block
Array of pointers to value list
Subscript conversion array
End position
Offset
Actual array
Value conversion array
Value list
Subscript conversion array
End position
Offset
Actual array
Smith
Ford
John
Jonny
Ken
New record No.
Record No.
View of personal name table
4100
(i)
(ii)
(iii)
(iv)
(v)
(vi)

FIG. 42

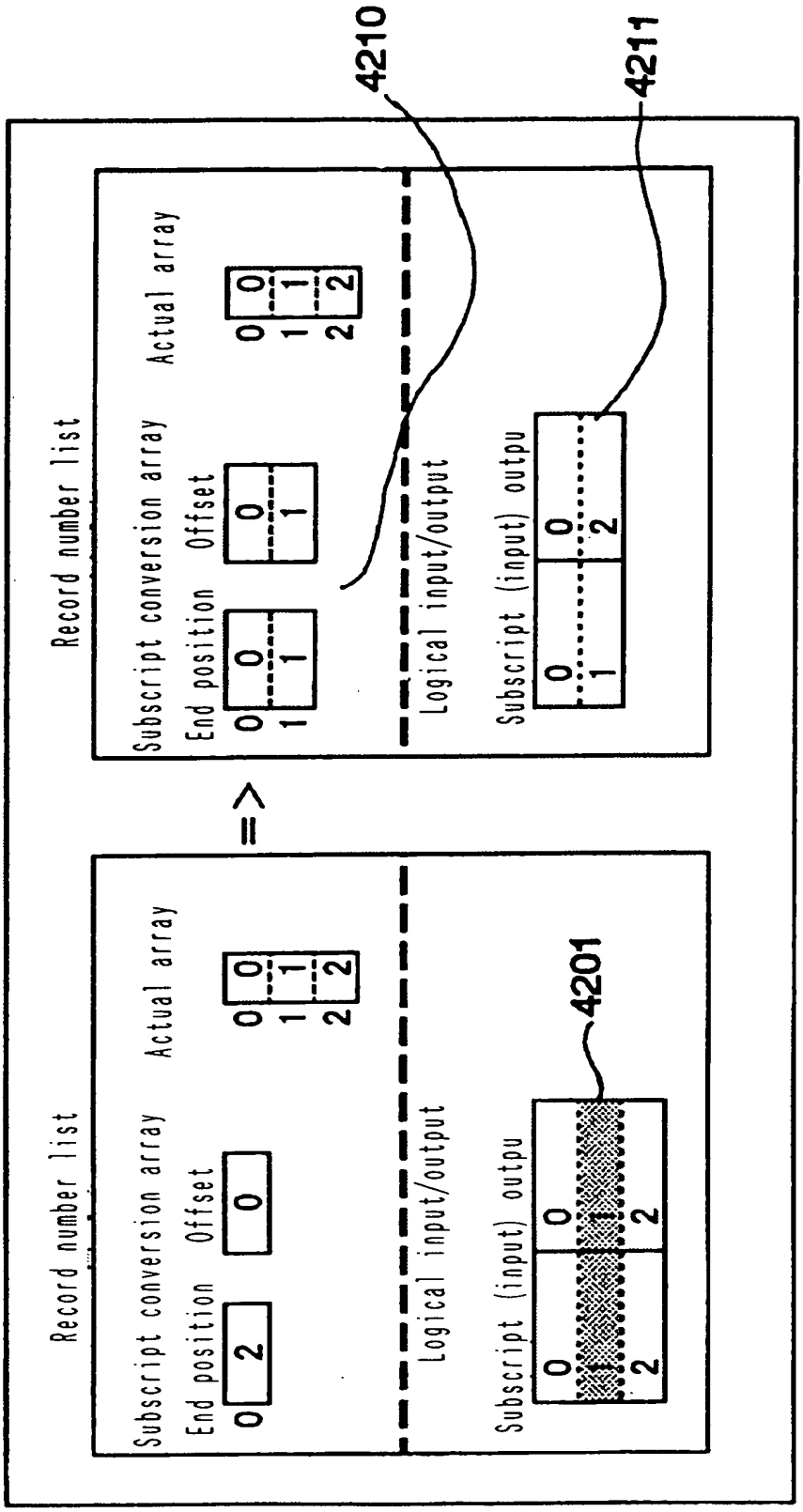
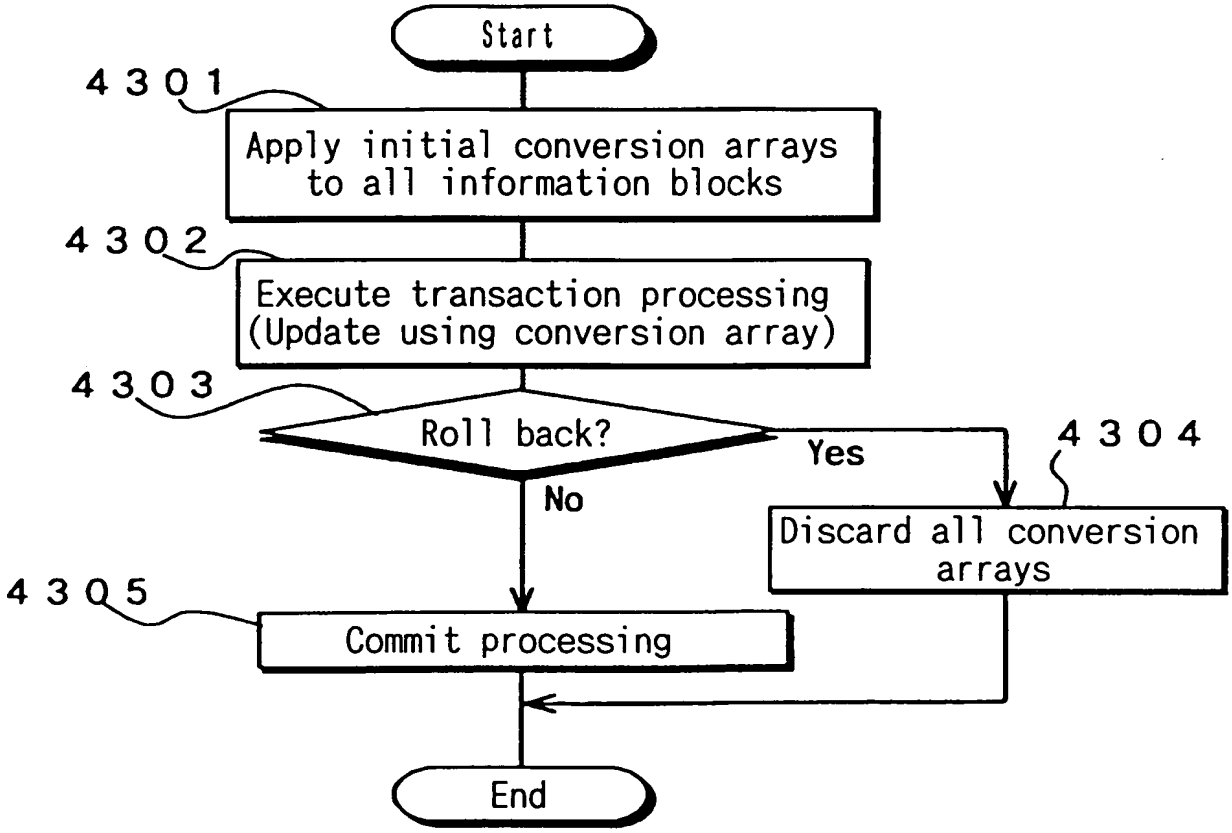

Record number list
Subscript conversion array
Actual array
End position
Offset
Logical input/output
Subscript (input) outpu
4201
4210
4211
=>

FIG. 43



FOR OFFICIAL USE ONLY

FIG. 44

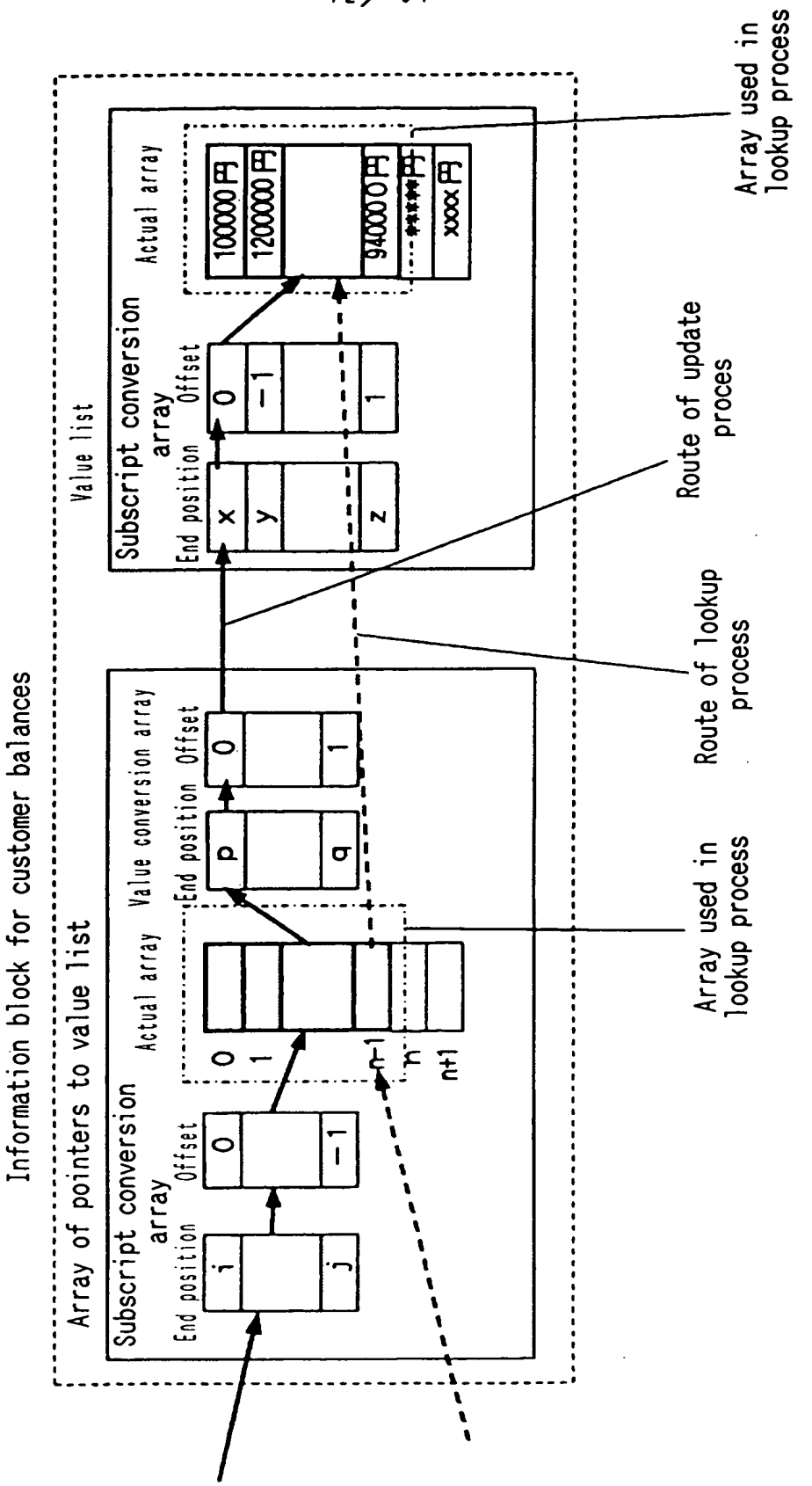

Information block for customer balances
Array of pointers to value list
Value list
Subscript conversion array
End position
Offset
Actual array
Value conversion array
i
j
0
-1
p
q
x
y
z
100000円
1200000円
9400000円
xxxx円
0
1
n-1
n
n+1
Array used in lookup process
Route of lookup process
Route of update proces
Array used in lookup process

FIG. 45

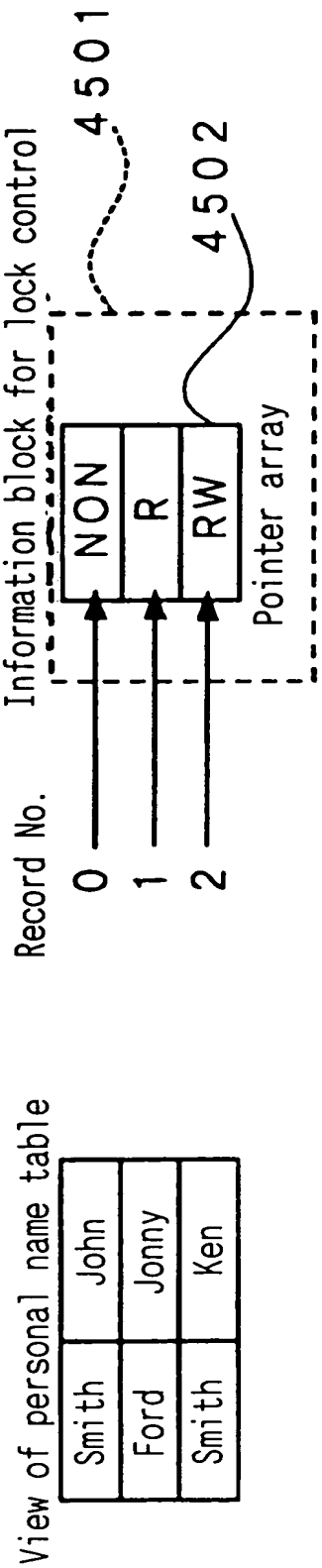

4501
4502
Information block for lock control
NON
R
RW
Pointer array
Record No.
0
1
2
View of personal name table
Smith
John
Ford
Jonny
Smith
Ken

FIG. 46A

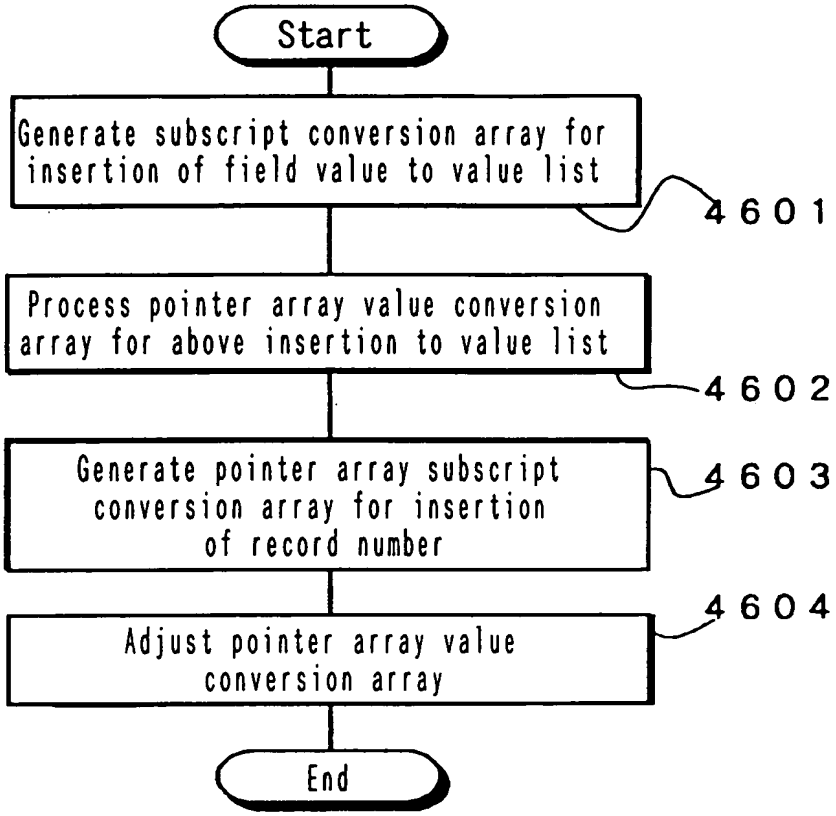


FIG. 46B

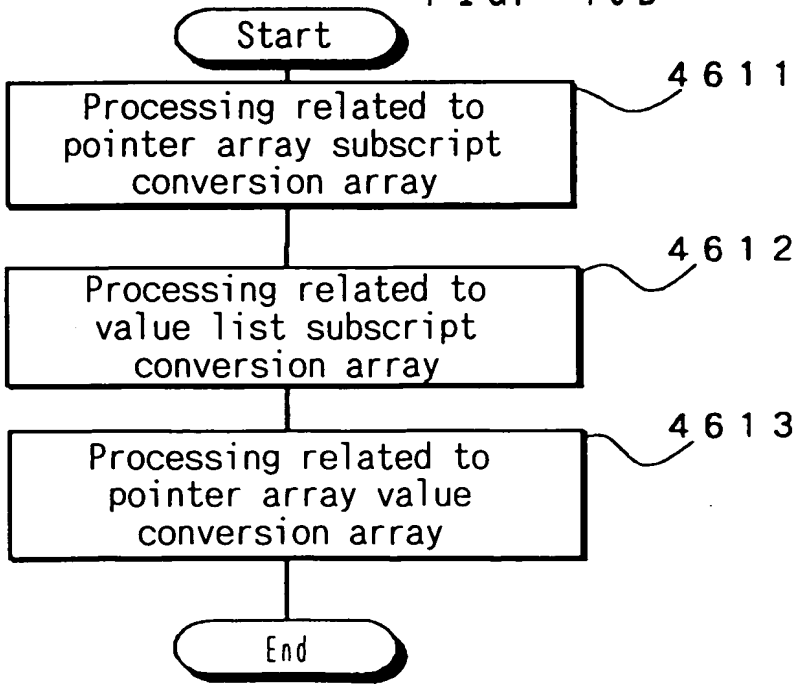


FIG. 47

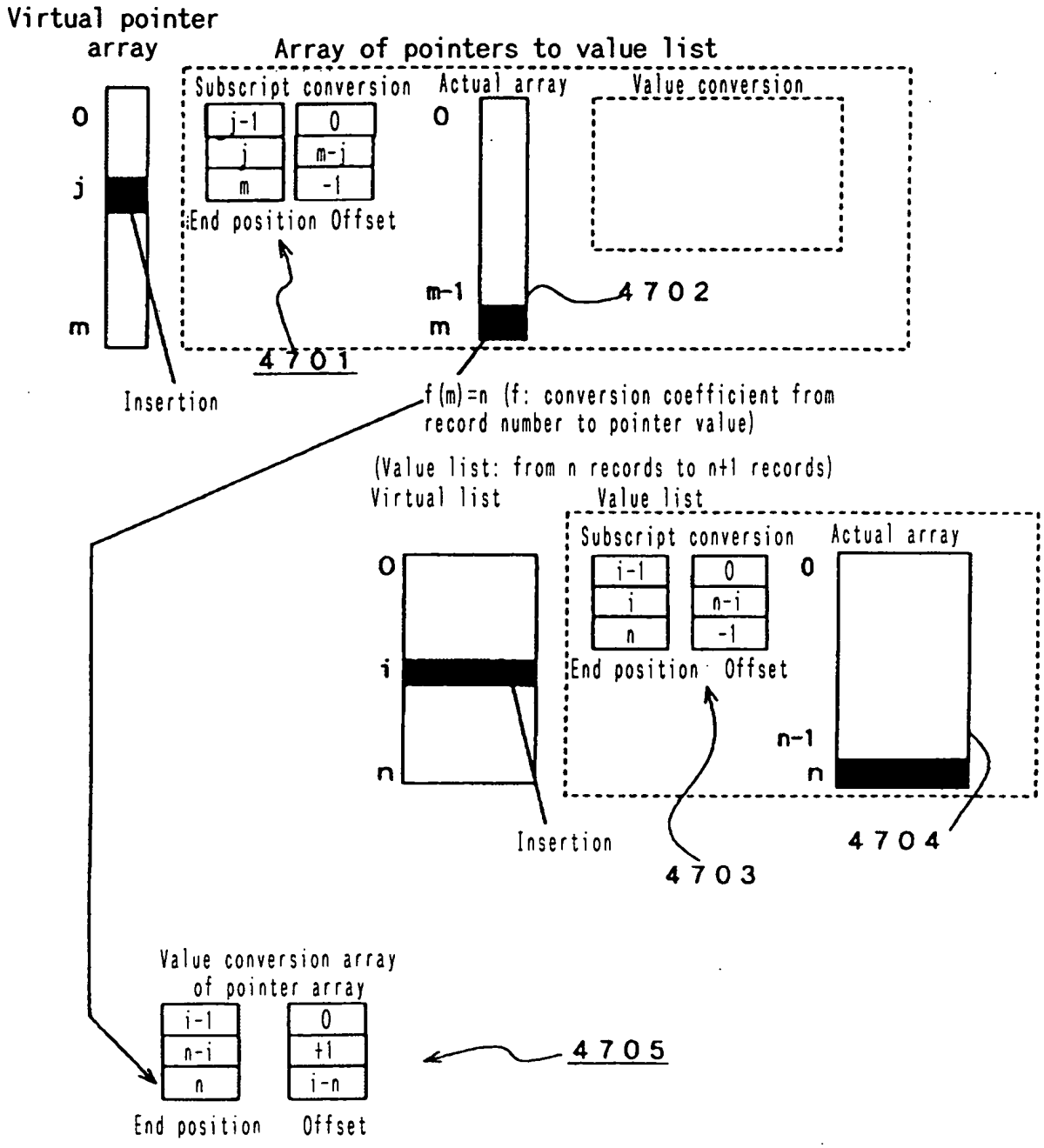

(Pointer array: from m records to m+1 records)
Virtual pointer array
Array of pointers to value list
Subscript conversion
Actual array
Value conversion
0
j
m
j-1
j
m
0
m-i
-1
End position
Offset
4701
m-1
m
4702
Insertion
f(m)=n (f: conversion coefficient from record number to pointer value)
(Value list: from n records to n+1 records)
Virtual list
Value list
Subscript conversion
Actual array
0
i
n
i-1
i
n
0
n-i
-1
End position
Offset
n-1
n
Insertion
4703
4704
Value conversion array of pointer array
i-1
n-i
n
0
+1
i-n
End position
Offset
4705

FIG. 48

(Pointer array: from m records to m-1 records)

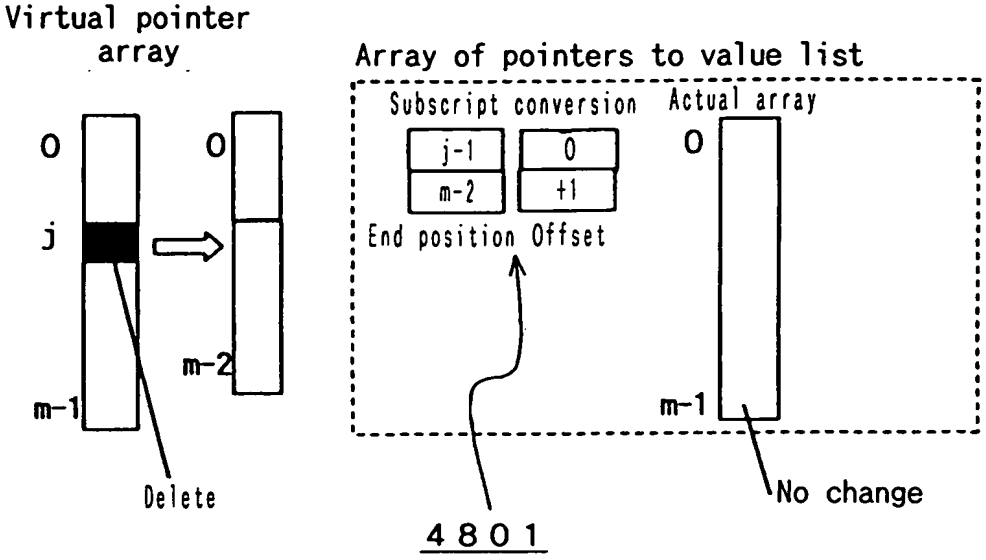


FIG. 49

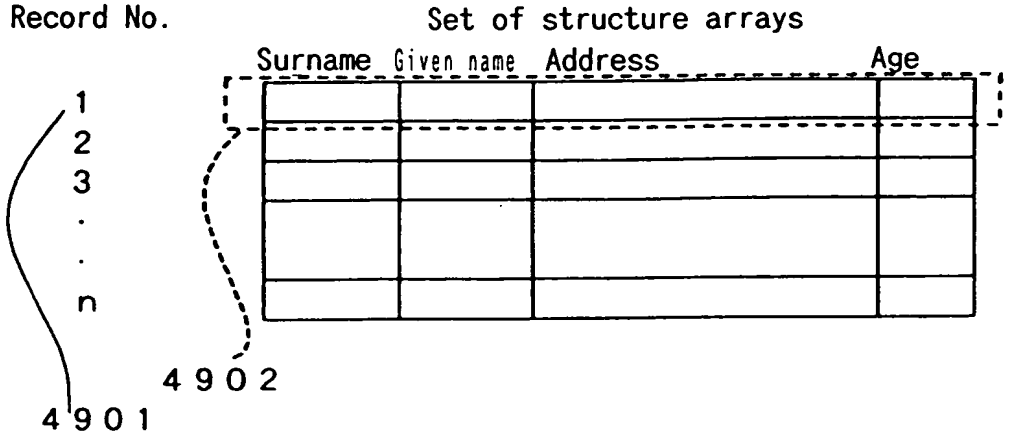


FIG. 50

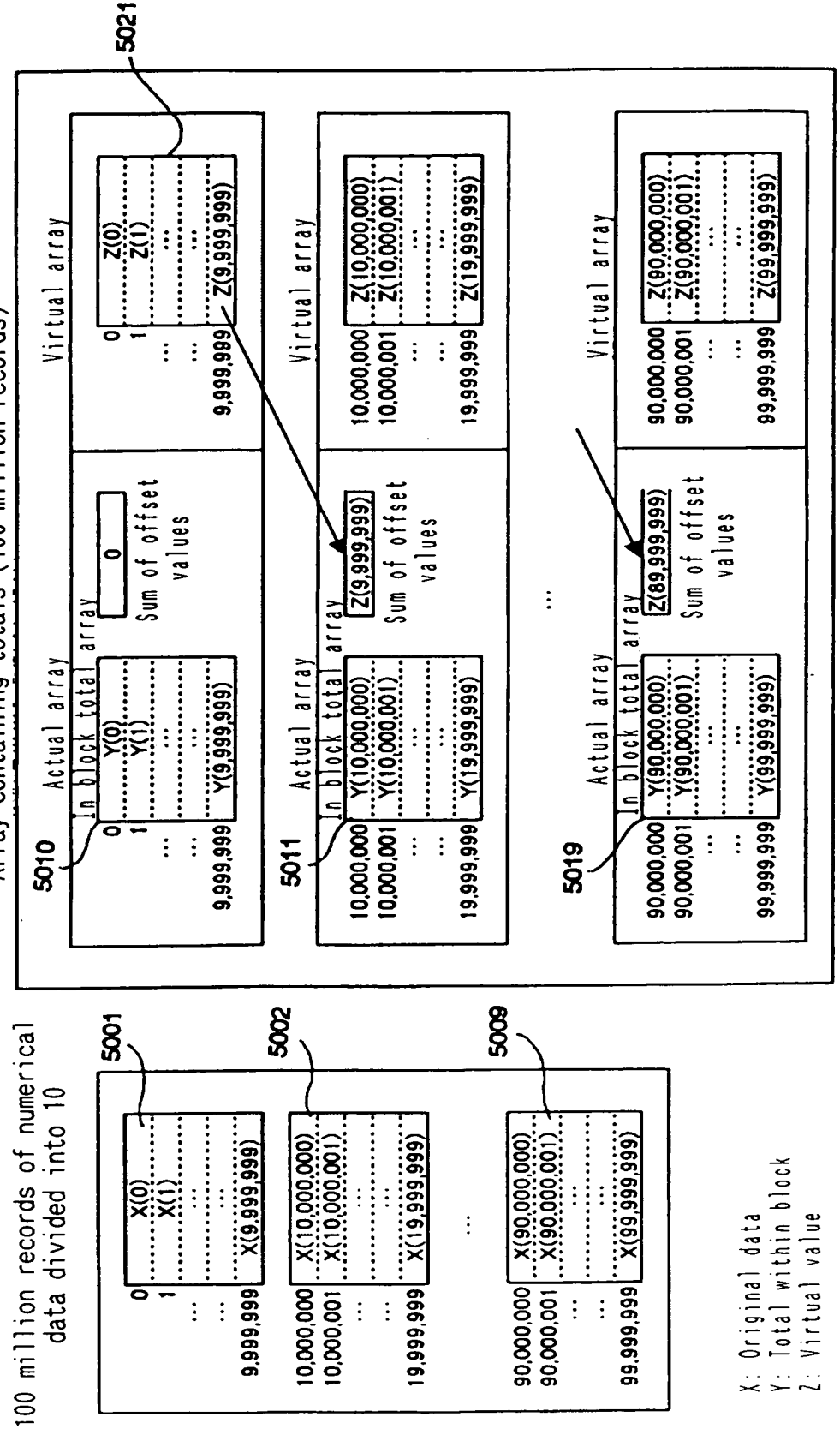

Array containing totals (100 million records)
5021
Virtual array
0
1
9,999,999
Z(0)
Z(1)
Z(9,999,999)
5010
Actual array
In block total array
Y(0)
Y(1)
Y(9,999,999)
0
Sum of offset values
5011
10,000,000
10,000,001
19,999,999
Z(10,000,000)
Z(10,000,001)
Z(19,999,999)
Z(9,999,999)
Y(10,000,000)
Y(10,000,001)
Y(19,999,999)
5019
90,000,000
90,000,001
99,999,999
Z(90,000,000)
Z(90,000,001)
Z(99,999,999)
Z(89,999,999)
Y(90,000,000)
Y(90,000,001)
Y(99,999,999)
100 million records of numerical data divided into 10
5001
5002
5009
X(0)
X(1)
X(9,999,999)
X(10,000,000)
X(10,000,001)
X(19,999,999)
X(90,000,000)
X(90,000,001)
X(99,999,999)
X: Original data
Y: Total within block
Z: Virtual value

FIG. 51

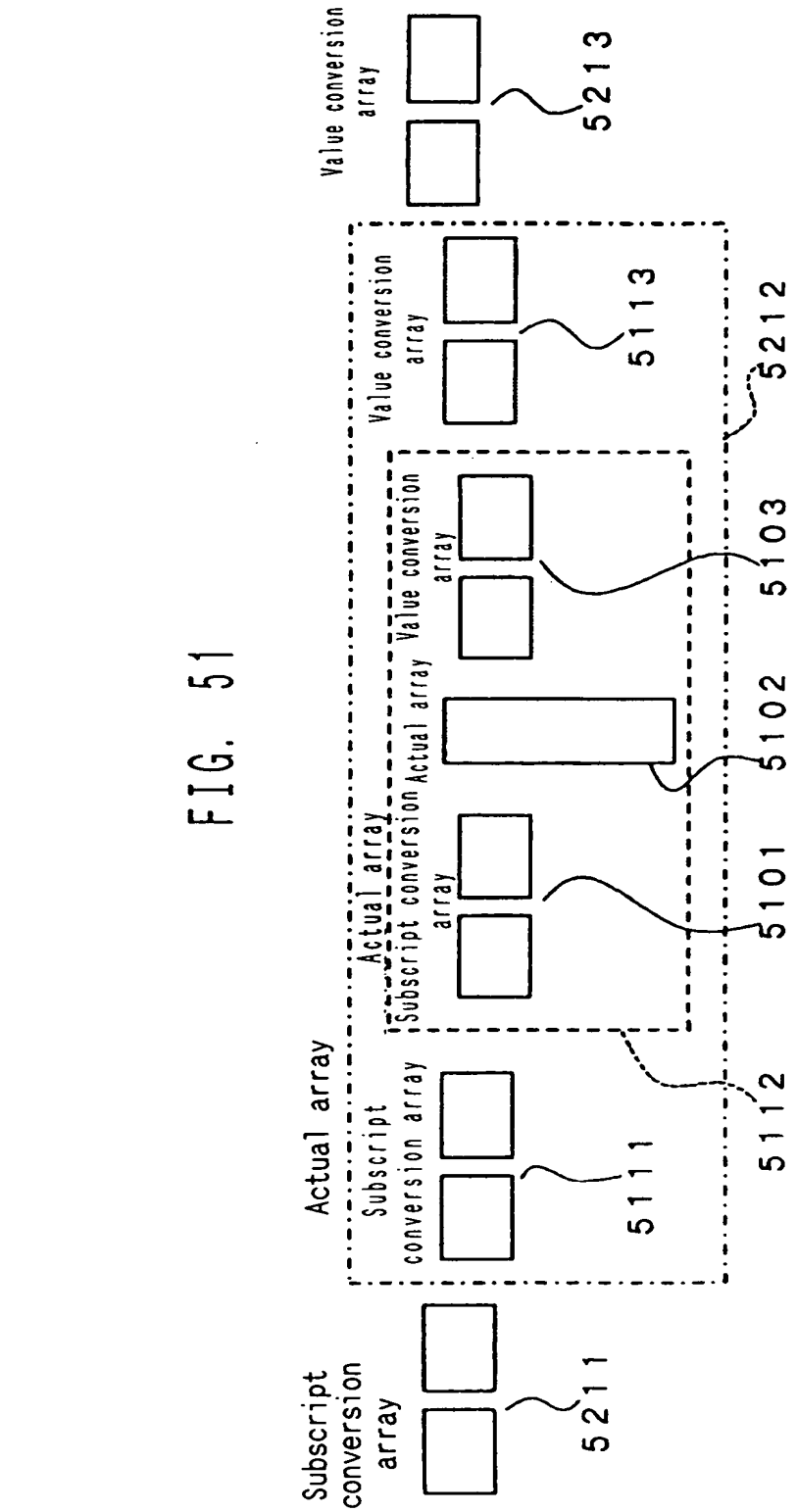

Subscript conversion array
5211
Actual array
Subscript conversion array
5111
5112
Actual array
Subscript conversion array
Actual array
Value conversion array
5101
5102
5103
Value conversion array
5113
5212
Value conversion array
5213

FIG. 52

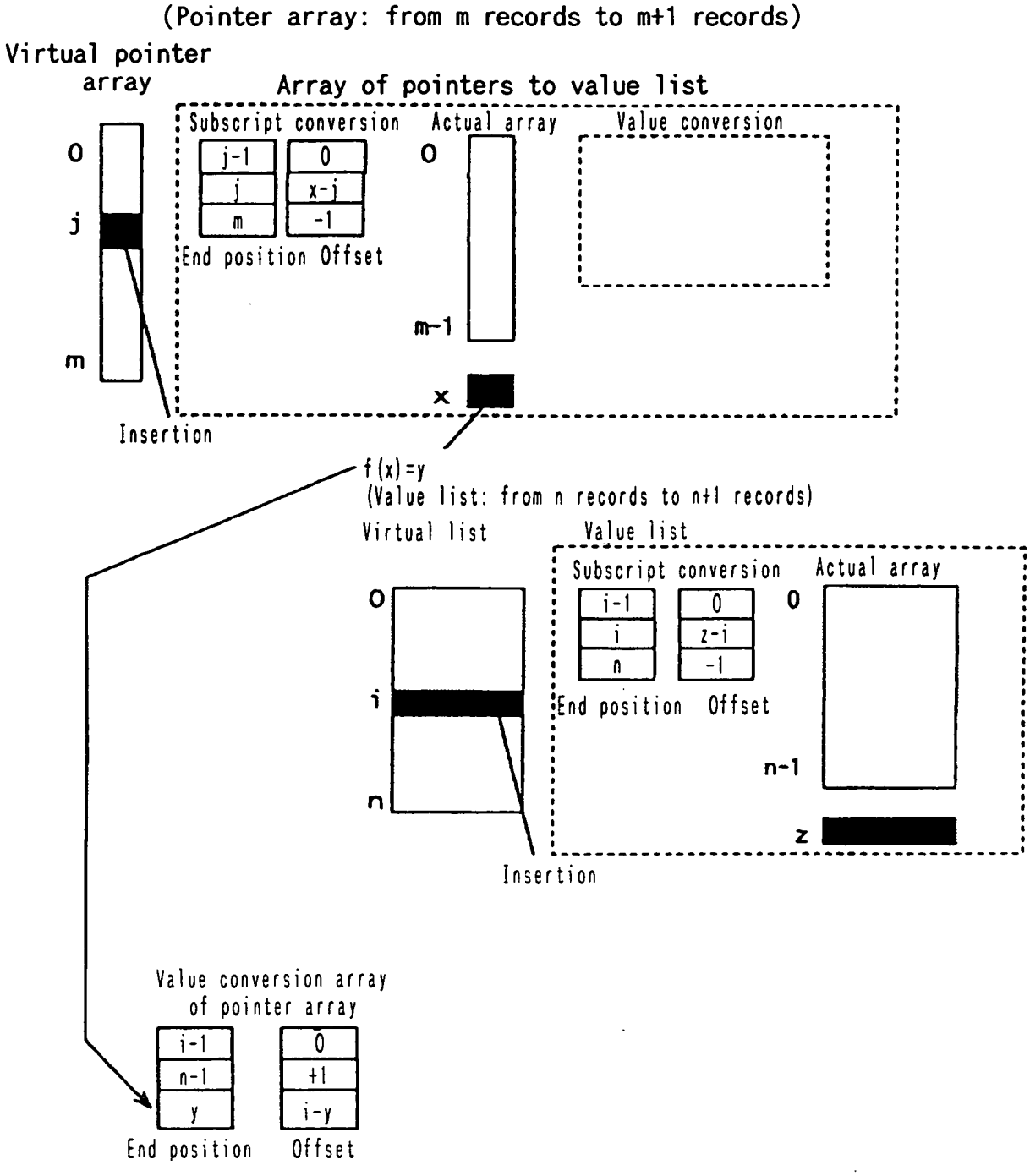

(Pointer array: from m records to m+1 records)
Virtual pointer array
Array of pointers to value list
Subscript conversion
Actual array
Value conversion
0
j
m
j-1
j
m
0
x-j
-1
End position
Offset
0
m-1
x
Insertion
f(x)=y
(Value list: from n records to n+1 records)
Virtual list
Value list
Subscript conversion
Actual array
0
i
n
i-1
i
n
0
z-i
-1
End position
Offset
n-1
z
Insertion
Value conversion array of pointer array
i-1
n-1
y
0
+1
i-y
End position
Offset

FIG. 53

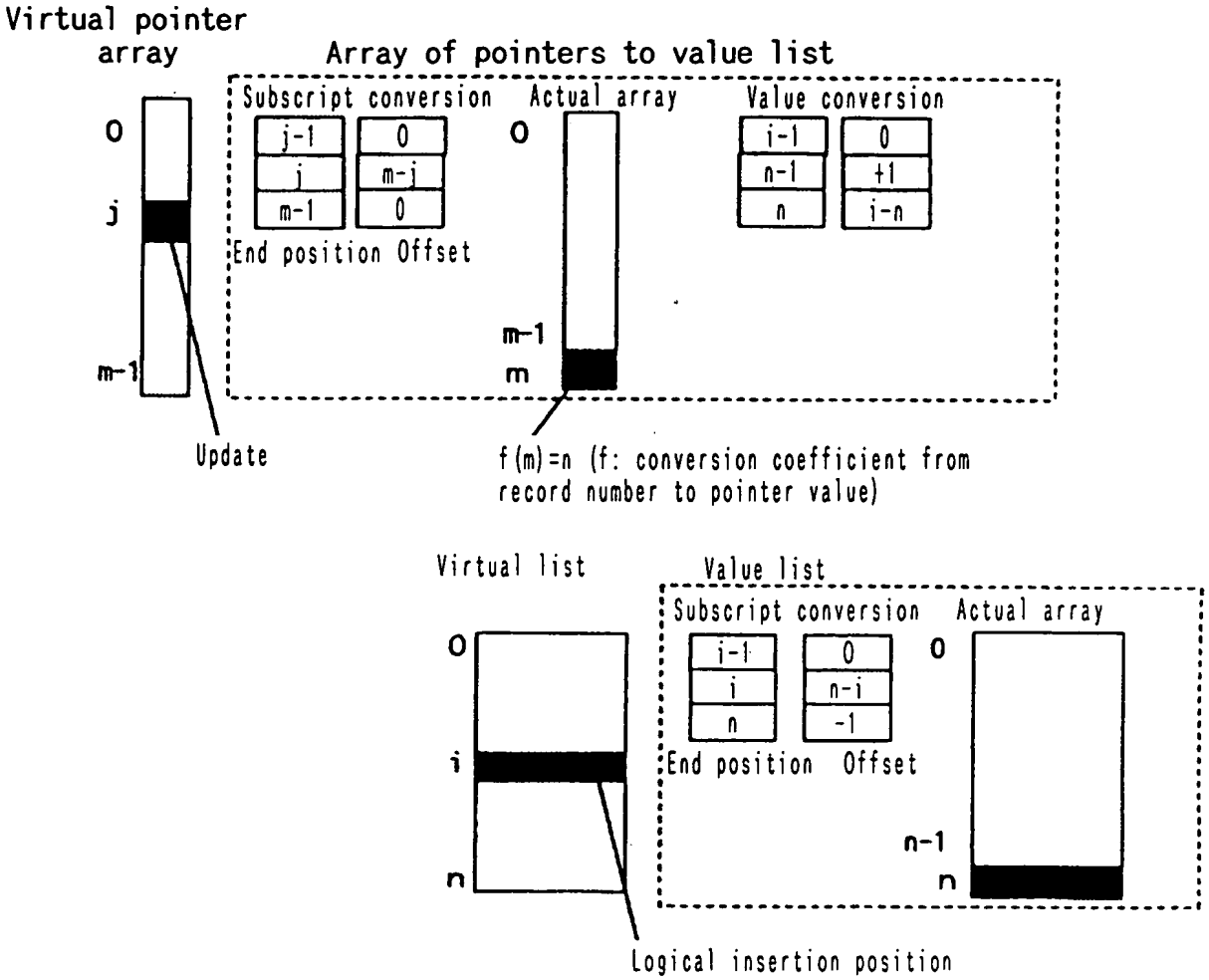
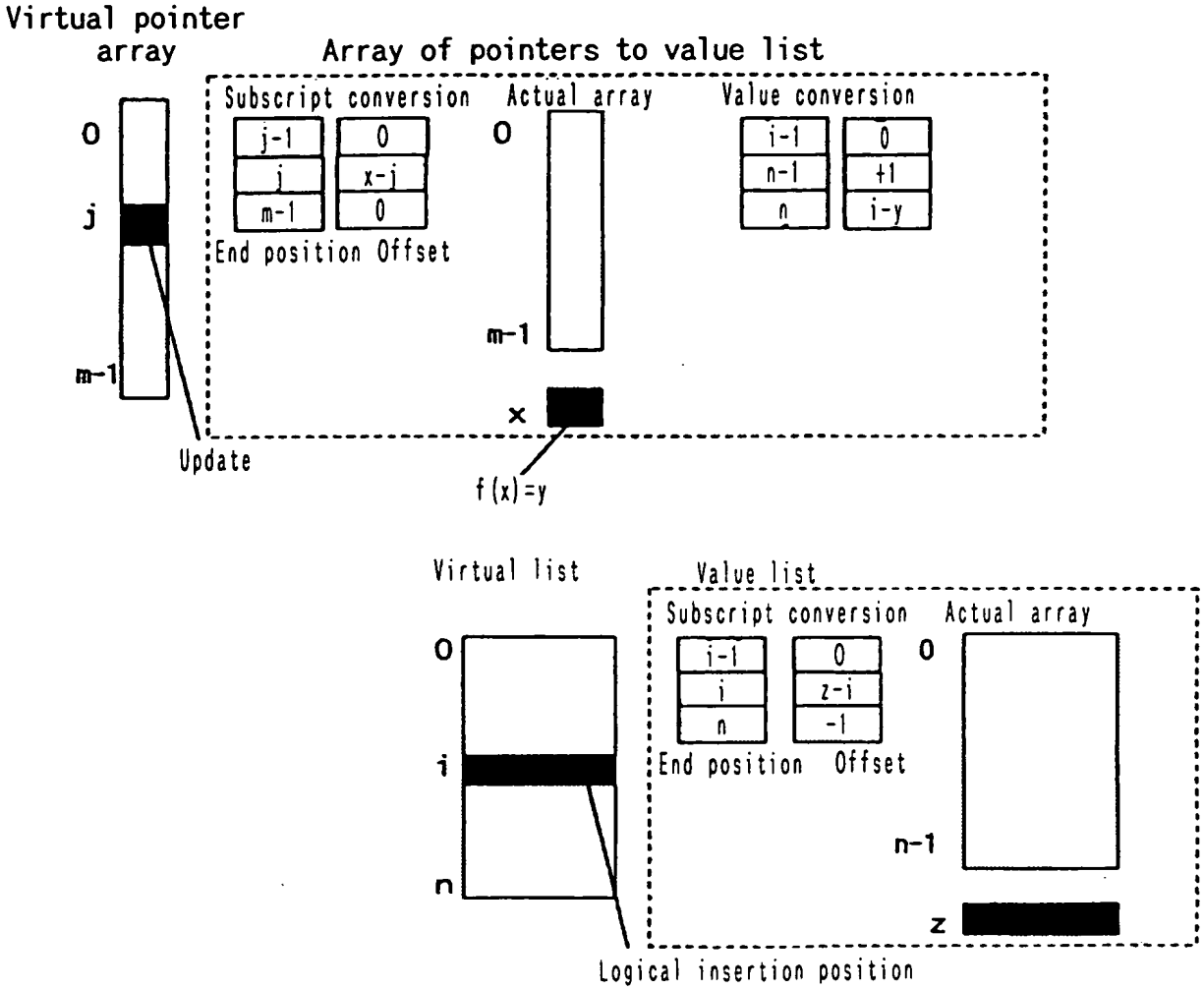

Virtual pointer array
Array of pointers to value list
Subscript conversion
Actual array
Value conversion
0
j
m-1
j-1
j
m-1
0
m-j
0
End position
Offset
i-1
n-1
n
0
+1
i-n
m-1
m
Update
f(m)=n (f: conversion coefficient from record number to pointer value)
Virtual list
Value list
Subscript conversion
Actual array
0
i
n
i-1
i
n
0
n-i
-1
End position
Offset
n-1
n
Logical insertion position

FIG. 54


Virtual pointer array
Array of pointers to value list
Subscript conversion
Actual array
Value conversion
0
j
m-1
j-1
j
m-1
End position
0
x-j
0
Offset
0
m-1
x
i-1
n-1
n
0
+1
i-y
Update
f(x)=y
Virtual list
Value list
Subscript conversion
Actual array
0
i
n
i-1
i
n
End position
0
z-i
-1
Offset
0
n-1
z
Logical insertion position